P. 1877/92

8 1992

informatyka

WYDAWNICTWO SIGMA NOT

Cena 18 000 zł

informatyka

nr 8 1992
Sierpień
Rok wydania XXVII

Miesięcznik
ISSN 0542-9951
INDEKS 36124

KOLEGIUM REDAKCYJNE:

mgr Jarosław DEMINET
mgr inż. Piotr FUGLEWICZ
mgr Teresa JABŁOŃSKA
(sekretarz redakcji)
Władysław KLEPACZ
(redaktor naczelny)
dr inż. Wojciech MOKRZYCKI
mgr inż. Jan RYŻKO
dr Zdzisław SZYJEWSKI
mgr Hanna WŁODARSKA –
– MARCZENKO

PRZEWODNICZĄCY
RADY PROGRAMOWEJ:

Prof. dr hab.
Juliusz Lech KULIKOWSKI

WYDAWCA:
Wydawnictwo Czasopism i Książek
Technicznych SIGMA NOT
Spółka z o.o.
ul. Biała 4
00-950 WARSZAWA
skrytka pocztowa 1004

Redakcja:
01-552 Warszawa,
Pl. Inwalidów 10, p. 104, 105
tel. 39-14-34

Materiałów nie zamówionych
redakcja nie zwraca

W sprawach ogłoszeń prosimy zwracać się bezpośrednio do Redakcji lub Działu Reklamy i Marketingu
00-950 Warszawa
ul. Mazowiecka 12
telefon: 27-43-65
telefaks: 20-31-16
teleks: 814550

W numerze:

Strona

Piąta Wiosenna Szkoła PTI. Od metodologii tworzenia systemów informatycznych do CASE – – *Zdzisław Szyjewski* 1
Informatyzacja sukcesem biznesu – *Wacław Iszkowski* 2
Inżynieria systemów informatycznych – od strategii do realizacji – *Teresa Piórkowska, Dariusz Puternicki, Witold Staniszkis* 9
AD/Cycle – IBM-owska filozofia inżynierii projektowania oprogramowania – *Krzysztof Madrjas* 18
Konkurs im. Jerzego Trybulskiego 27

W najbliższych numerach:

We wrześniowym numerze opublikujemy najciekawsze materiały przygotowane na IV Wyższą Górską Szkołę PTI, o tematyce „Współczesne obszary zastosowań informatyki", która odbyła się w Szczyrku w dniach 22–26 czerwca 1992 r.

- Jerzy Skrzypek i Mariusz Szubra prezentują w swym artykule gry decyzyjne jako narzędzie aktywnego kształcenia menedżerów.
- Marek Wawrzyńczyk i Katarzyna Stąpor opisują wpływ algorytmów tablic rozproszonych na efektywność działania systemów ekspertowych.
- Danuta Kosmowska-Miszalska charakteryzuje zintegrowany pakiet GURU – uniwersalne narzędzie do tworzenia regułowych systemów ekspertowych i na jego podstawie omawia strojenie mechanizmu rozumowania systemów ekspertowych pod kątem różnych zastosowań.
- Krzysztof Michalik i Zbigniew Twardowski przedstawiają system PC-SHELL stanowiący narzędzie do budowy dziedzinowych systemów ekspertowych.

Warunki prenumeraty

Przyjęcie prenumeraty – wyłącznie na podstawie dokonanej wpłaty na drukach dostarczanych dotychczasowym prenumeratorom przez Wydawnictwo, lub nowym – po uprzednim zgłoszeniu zapotrzebowania (pisemnie lub telefonicznie) w Zakładzie Kolportażu Wydawnictwa.

Blankiet wpłaty – powinien zawierać następujące informacje: dokładna nazwa i adres (z kodem pocztowym) zamawiającego, tytuły zamawianych czasopism, ich liczbę i okres prenumeraty.

Wpłata – zgodnie z podanymi cenami należy dokonać w banku lub w UPT na konto podane na naszym blankiecie, tj:
Państwowy Bank Kredytowy III O/Warszawa nr: 370015-1573-139-11

Prenumeratorzy zbiorowi – osoby prawne obowiązują blankiety „Wpłata-Zamówienie". Cena normalna.

Prenumeratorzy indywidualni – osoby fizyczne obowiązują blankiety typu przekazy dla wpłat na rachunki bankowe. Cena normalna.

Prenumerata ulgowa – zgodnie z podaną ceną ulgową przysługuje wyłącznie osobom fizycznym, będącym członkami SNT, studentom i uczniom szkół zawodowych. Uczniowie szkół ogólnokształcących mogą zamówić w prenumeracie ulgowej tylko miesięcznik „Aura"

Uwaga! w podanym okresie prenumeraty można zamówić tylko po jednym egzemplarzu z każdego tytułu.

Prenumerata ze zleceniem wysyłki za granicę – cena prenumeraty ze zleceniem wysyłki za granicę jest dwukrotnie wyższa od ceny normalnej.

Należy podać dokładny adres odbiorcy za granicą.

Terminy przyjmowania prenumeraty:
– do 10 listopada na I, II, III, IV kwartał następnego roku
– do 28 lutego na II, III, IV kwartał br.
– do 31 maja na III i IV kwartał br.
– do 31 sierpnia na IV kwartał br.

Zmiany w prenumeracie, np. zmiana liczby tytułów, liczby egzemplarzy, rezygnacja z prenumeraty, można zgłaszać tylko w podanych terminach z mocą obowiązującą od następnego kwartału.

Egzemplarze archiwalne (z lat ubiegłych)
Można nabyć za gotówkę w Klubie Prasy Technicznej, Warszawa, ul. Mazowiecka 12 (tel. 26-80-16) lub zamówić pisemnie w Zakładzie Kolportażu, Dział Handlowy, 00-950 Warszawa, skr. poczt. 1004 (tel. 40-37-31), na rachunek lub za zaliczeniem pocztowym.

Informacji o prenumeracie udziela: Zakład Kolportażu Wydawnictwa SIGMA-NOT Spółka z o.o., 00-716 Warszawa, ul. Bartycka 20, skr. 1004. Telefony: 40-00-21 wewn. 293, 295, 299 lub 40-30-86, 40-35-89.

Wstępna cena jednego egzemplarza na 1992 rok: normalna – 18 000 zł, ulgowa – 13 500zł

Wartość prenumeraty (w zł):
Normalna: kwartalna – 54 000, półroczna 108 000, roczna 216 000
Ulgowa: kwartalna – 40 500, półroczna 81 000, roczna 162 000

Uwaga: W przypadku zmiany cen w okresie objętym prenumeratą, prenumeratorzy zobowiązani są do dopłaty różnicy cen.

Skład i druk: Drukarnia SIGMA NOT Sp. z o.o.

POLITECHNIKA GDAŃSKA Biblioteka Główna w Gdańsku
P. 1877/92

Piąta Wiosenna Szkoła PTI

Od metodologii tworzenia systemów informatycznych do CASE

Piąta Wiosenna Szkoła Polskiego Towarzystwa Informatycznego zamyka cykl monotematycznych, tygodniowych spotkań informatyków na tematy związane z metodyką tworzenia systemów informatycznych. Majowe spotkania członków PTI mają już swoją tradycję i stałych bywalców.

Pierwsze majowe spotkanie w roku 1988 było wynikiem autentycznej potrzeby środowiska informatyków, które odczuwało konieczność wymiany poglądów na temat metodyk tworzenia systemów informatycznych. W uporządkowaniu problemów metodycznych upatrywano sposób na sprawniejsze i lepsze jakościowo wytwarzanie systemów informatycznych. W trakcie Pierwszej Wiosennej Szkoły PTI dokonano podsumowania poglądów różnych środowisk na problemy metodyk wytwarzania systemów informatycznych.

Druga Wiosenna Szkoła PTI skoncentrowała się na metodach, technikach i narzędziach tworzenia systemów informatycznych, dając przegląd ich praktycznego stosowania oraz informacje o nowościach światowych. Niektóre referaty pozwalały inaczej spojrzeć na znane od dawna i stosowane metody i techniki.

Trzecia Wiosenna Szkoła PTI wychodziła naprzeciw tendencjom, sygnalizowanym na dwóch wcześniejszych spotkaniach. Referaty tej Szkoły dotyczyły komputerowego wspomagania tworzenia systemów informatycznych. Obrady Szkoły koncentrowały się głównie wokół informacji o postępach i doświadczeniach w zakresie CASE w innych krajach. Jedno polskie wykorzystanie prostego pakietu, przedstawione w trakcie Szkoły, było jaskółką, która dopiero miała uczynić wiosnę. Wymiana poglądów była twórcza, chociaż wykazała duży dystans między warsztatem pracy informatyka w Polsce a stosowanym w innych krajach komputerowo wspomaganym wytwarzaniem systemów.

Kolejne edycje Wiosennej Szkoły PTI zawsze starały się poruszać problematykę aktualnie najbardziej interesującą środowisko informatyczne, uzgadniając tematykę Szkół z organizatorami Górskich Szkół PTI. Wydaje się, że działania te były skuteczne, o czym może świadczyć frekwencja na kolejnych Szkołach.

Problemem początku lat dziewięćdziesiątych w Polsce stały się sieci mikrokomputerowe, które miały efektywnie spiąć moce licznych PC-tów, dając dodatkową wartość na miarę naszych możliwości. W referatach Czwartej Wiosennej Szkoły PTI przedstawiono problemy i rozwiązania w zakresie lokalnych sieci mikrokomputerowych. Zasygnalizowano problemy sieci rozległych, dzieląc się kilkuletnimi doświadczeniami zespołu, który pracował nad krajową siecią rozległą. Tematyka sieci, a głównie doświadczenia ich stosowania, obrazowały nowe warunki wytwarzania aplikacji.

Tematyka czterech Szkół zmierzała do podjęcia problematyki, która od dawna była celem organizatorów. Chcieliśmy zaprezentować konkretne narzędzia wspomagające bardzo efektywnie wytwarzanie systemów informatycznych. Stosowane w naszej praktyce prymitywne narzędzia wytwarzania powodowały frustracje informatyków i użytkowników tworzonych aplikacji. Wydaje się, że nadszedł czas zmian jakościowych, które umożliwią wykorzystanie wiedzy informatyków zgodnie z oczekiwaniami. Chcemy pokazać aktualne, dostępne na naszym rynku możliwości i przedyskutować w fachowym gronie szanse i zagrożenia tej koniecznej zmiany jakościowej. O potrzebie takich działań świadczą liczne inicjatywy i wypowiedzi, które mówią o systemach otwartych. System otwarty to pewna klasa produktu, na który istnieje zapotrzebowanie, a my jako informatycy powinniśmy zastanawiać się jak tworzyć takie systemy. Obecna Szkoła powinna stanowić element tych rozważań.

Piąta Wiosenna Szkoła PTI spina tematykę dotychczasowych Szkół próbując przedstawić nowe podejście do wytwarzania systemów informatycznych. Nowe podejście metodyczne charakteryzuje się wykorzystaniem komputerowego wspomagania w całym cyklu życia systemu. Powstało środowisko sprzętowo–programowe wspomagające wytwarzanie systemu, pozwalając informatykom skoncentrować się na rozwiązywaniu problemów, a nie na zwalczaniu trudności technicznych czy technologicznych. Języki czwartej generacji, narzędzia CASE, systemy zarządzania bazami danych, możliwości i opcje systemów operacyjnych, to podstawowe elementy nowego środowiska, które wspomaga szybkie i sprawne wytwarzanie systemów wysokiej jakości.

Gospodarzem pierwszego dnia tegorocznej imprezy była firma **Digital Equipment Polska,** jeden ze sponsorów Szkoły. W wystąpieniach swoich pracowników firma przedstawiła nową metodyczną koncepcję tworzenia systemów informatycznych oraz środowisko CASE, wspomagające wytwarzanie systemów o nazwie COHESION. Wykłady były wspomagane pokazami narzędzi CASE na stacji roboczej produkcji firmy Digital. W uzupełnieniu, przedstawiciel firmy CasePol, partnera Digitala, przedstawił zintegrowane środowisko CASE, użytkowane na sprzęcie Digitala.

W drugim dniu gospodarzem był następny sponsor Szkoły – firma **CSBI (Computer Systems for Business International LTD.).** Wystąpienia pracowników CSBI koncentrowały się wokół języka PROGRESS oraz narzędzi typu CASE, wywodzących się z tego języka lub współpracujących z tym językiem.

W trzecim dniu odbyły się zajęcia praktyczne prowadzone przez zespół z Uniwersytetu Gdańskiego. Był to jednodniowy tutorial na temat: „Metody analizy i projektowania systemów w praktyce" (wykłady, ćwiczenia, analiza przypadków, praca w grupach, dyskusje).

Czwarty dzień był obsługiwany również przez sponsorów – firmy ZETO-RODAN oraz ASCOM. Przed południem **dr Witold Staniszkis,** z ZETO-RODAN przedstawił, referat **„Inżynieria systemów informatycznych – od strategii do realizacji".** Po południu firma ASCOM przedstawiła ofertę produktów oferowanych przez nią w Polsce.

Piąty dzień Szkoły był obsługiwany przez następnego sponsora Szkoły – firmę IBM Polska, której przedstawiciel wygłosił referat **„AD/Cycle – IBM-owska filozofia inżynierii projektowania oprogramowania".**

Pięć lat, jakie minęły od Pierwszej Wiosennej Szkoły PTI w 1988 r., to bardzo burzliwy okres w historii Polski. W tym czasie przechodziliśmy rewolucyjne zmiany jakościowe i ilościowe również w informatyce. Widać to porównując Pierwszą i Piątą Szkołę. Pierwsze Szkoły miały charakter wyłącznie akademicki, co oznacza, że większość wykładowców reprezentowała środowisko akademickie, a referaty najczęściej sumowały, oceniały i wytyczały kierunki zalecanych działań. W kolejnych edycjach Szkoły zaczęły pojawiać się produkty handlowe, które pokazywały nowe efektywne sposoby wytwarzania systemów. Przedstawiona lista firm, prezenterów Piątej Wiosennej Szkoły wyraźnie tę tezę potwierdza. Takie są realia obecnych czasów i konferencje, dotychczas naukowe, muszą to uwzględniać.

Mimo, że staramy się zachować formułę organizacyjną Szkoły, to pewne zmiany organizacyjne są konieczne i wynikają z nie tylko naukowego charakteru Szkoły. Piąta Wiosenna Szkoła PTI została zrealizowana w dwóch współbieżnych warstwach: naukowo-merytorycznej i promocyjno-handlowej. Wprowadzenie w program Szkoły warstwy komercyjnej pozwoliło znacznie obniżyć koszty uczestnictwa. Mamy nadzieję, że innowację tę uczestnicy przyjmą ze zrozumieniem, gdyż uatrakcyjni to również program Szkoły. Staraliśmy się, żeby firmy przygotowujące wystąpienia merytoryczne były równocześnie sponsorami Szkoły, co sprawiło, że wystąpienia promocyjno-handlowe w dużej części dotyczyły rozwiązań merytorycznie związanych z tematyką Szkoły.

ZDZISŁAW SZYJEWSKI

Artykuł ten został opracowany dla kwartalnika DEC-forum i za zgodą Redakcji z Digital Equipment Polska był publikowany w materiałach V Wiosennej Szkoły PTI – CASE'92 w Świnoujściu.

WACŁAW ISZKOWSKI
Digital Equipment Polska
Warszawa

Informatyzacja sukcesem biznesu

Część Czytelników pamięta zapewne książkę i hasło – „Informatyka kluczem do dobrobytu" [1]. Pozostawmy na boku weryfikację treści tej książki, pisanej w innej epoce i w innej sytuacji. Obecnie jednak sprawdza się główne hasło, gdyż właśnie teraz jedynie dzięki informatyce będzie możliwy dalszy rozwój gospodarczy. I co ważniejsze, nie dotyczy to tylko Polski, ale przede wszystkim USA i reszty nowoczesnego świata. W tym eseju spróbujemy powiązać tę tezę z aktualnymi możliwościami informatyki.

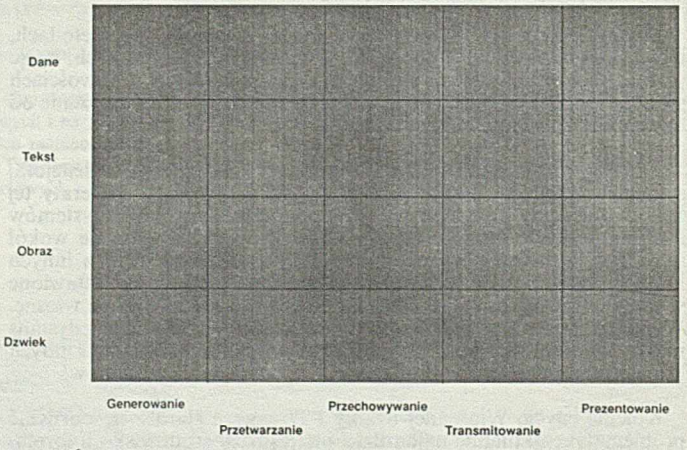
Agresja informatyki

Informatyzacja staje się ogólnoświatowym pojęciem, gdyż nawet w językach angielskim i anglo-amerykańskim pojawił się termin *informationalization*[2]. Dlaczego zajęto się hasłem ogólniejszym niż komputeryzacja? Dlaczego nawet Prezydent Bush dał przykład swoim rodakom, ucząc się obsługi komputera? Dlaczego pomimo coraz mniejszych zysków, czołowe firmy komputerowe inwestują ogromne pieniądze w badania nad nowymi technologiami i projektami systemów?

Na te wszystkie pytania można dać jedną odpowiedź. Informatyka osiągnęła wiek dojrzały i zażądała prawa decydowania o losach świata. Z drugiej strony gospodarka światowa w dotychczasowej formie staje się przestarzała. Era dominacji przemysłu ze wszelkimi tego dobrymi i złymi skutkami przemija. Jedynie zmiana jakościowa ekonomii i organizacji gospodarki oraz społeczeństwa może pozytywnie wpłynąć na dalszy rozwój świata. Informatyka agresywnie zdobywa nowe pola wpływów i zaczyna dominować we wszystkich dziedzinach życia. Wielu ludzi, szczególnie starszych, fakt ten nieco przeraża.

KOMPUTER	*– to każdy wie, co to jest;*
KOMPUTERYZACJA	*– zakup i użytkowanie komputera;*
INFORMATYKA	*– technologia przetwarzania informacji;*
INFORMATYZACJA	*– zakup i wykorzystywanie informatyki.*

Zanim jednak spróbujemy przeanalizować ofertę informatyki – spójrzmy nieco na poprzednie lata. Człowiek w swoim środowisku posługuje się czterema formami przekazu informacji: dźwięk, obraz, pismo i inne dane, a w szczególności liczby (pozostałe zmysły człowieka – smak, dotyk i zapach być może będą wykorzystane do przekazywania informacji dopiero w przyszłości). Równocześnie każda z tych czterech form jest generowana, przetwarzana, przechowywana, transmitowana oraz prezentowana. Przedstawmy to w postaci tabeli (rys. 1), w której wiersze identyfikowane są formami, a kolumny metodą obsługi informacji. A teraz wpiszmy w tabelę (rys. 2) znane w początkach lat 60. urządzenia techniczne, starając się uwypuklić ich charakterystyczne cechy formy i obsługi informacji.



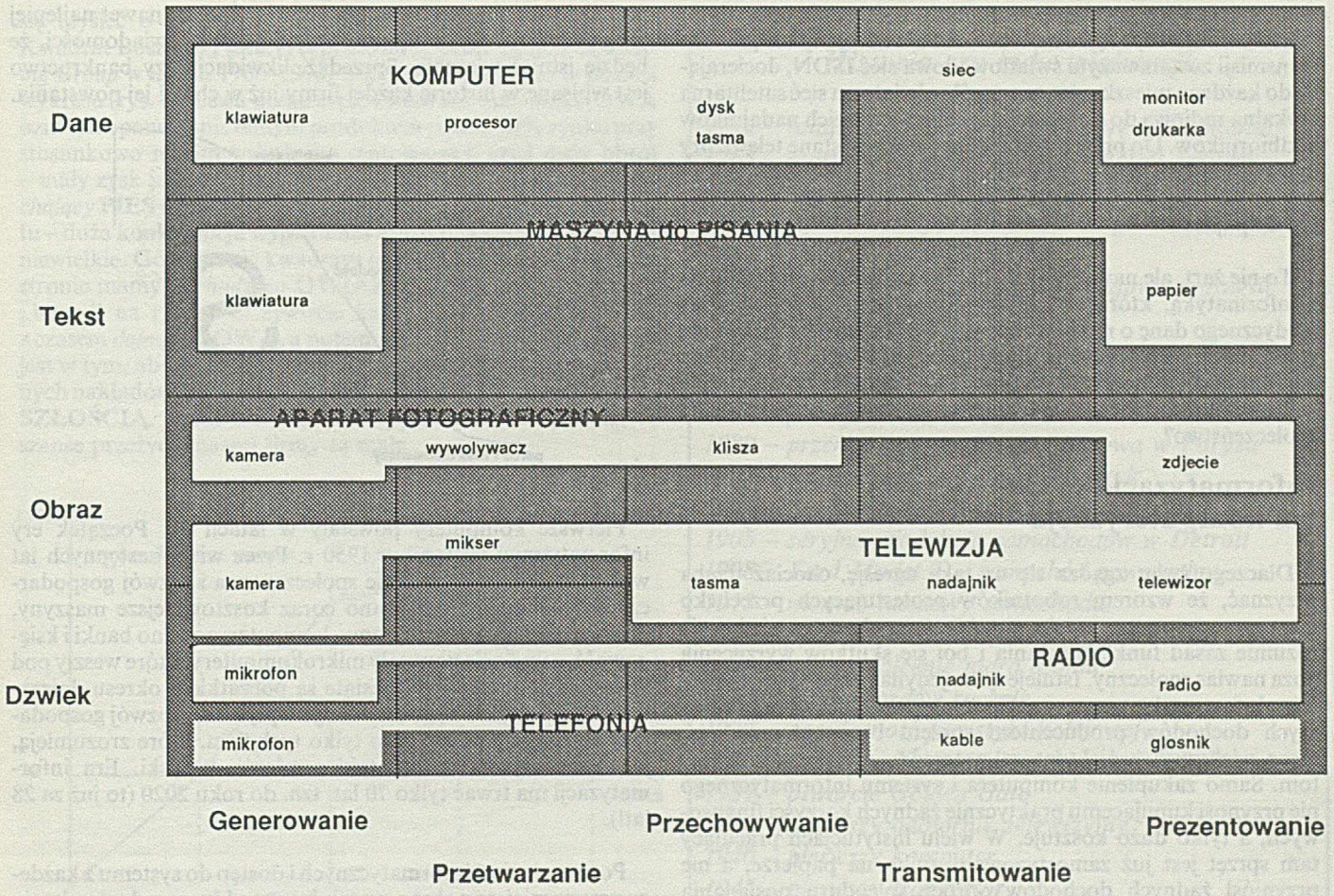
Rys. 1. Środki przekazu informacji

Na przykład radio: generowany dźwięk chwytany jest przez mikrofon, a następnie jest nieco przetwarzany (modulowany) i transmitowany z nadajnika na falach do odbiorników. Dodając urządzenia magnetofonowe, dźwięk możemy przechowywać na taśmach. Inne urządzenie – telefon – powstało w momencie wynalezienia sposobu przesyłania dźwięku drutem. Podobnie możemy przeanalizować takie urządzenia, jak: maszynę do pisania, aparat fotograficzny, telewizję i wreszcie komputery – przeznaczone do obliczeń. Proszę zauważyć, że każde z tych urządzeń oraz związane z nimi infrastruktury były zarządzane przez różne firmy.

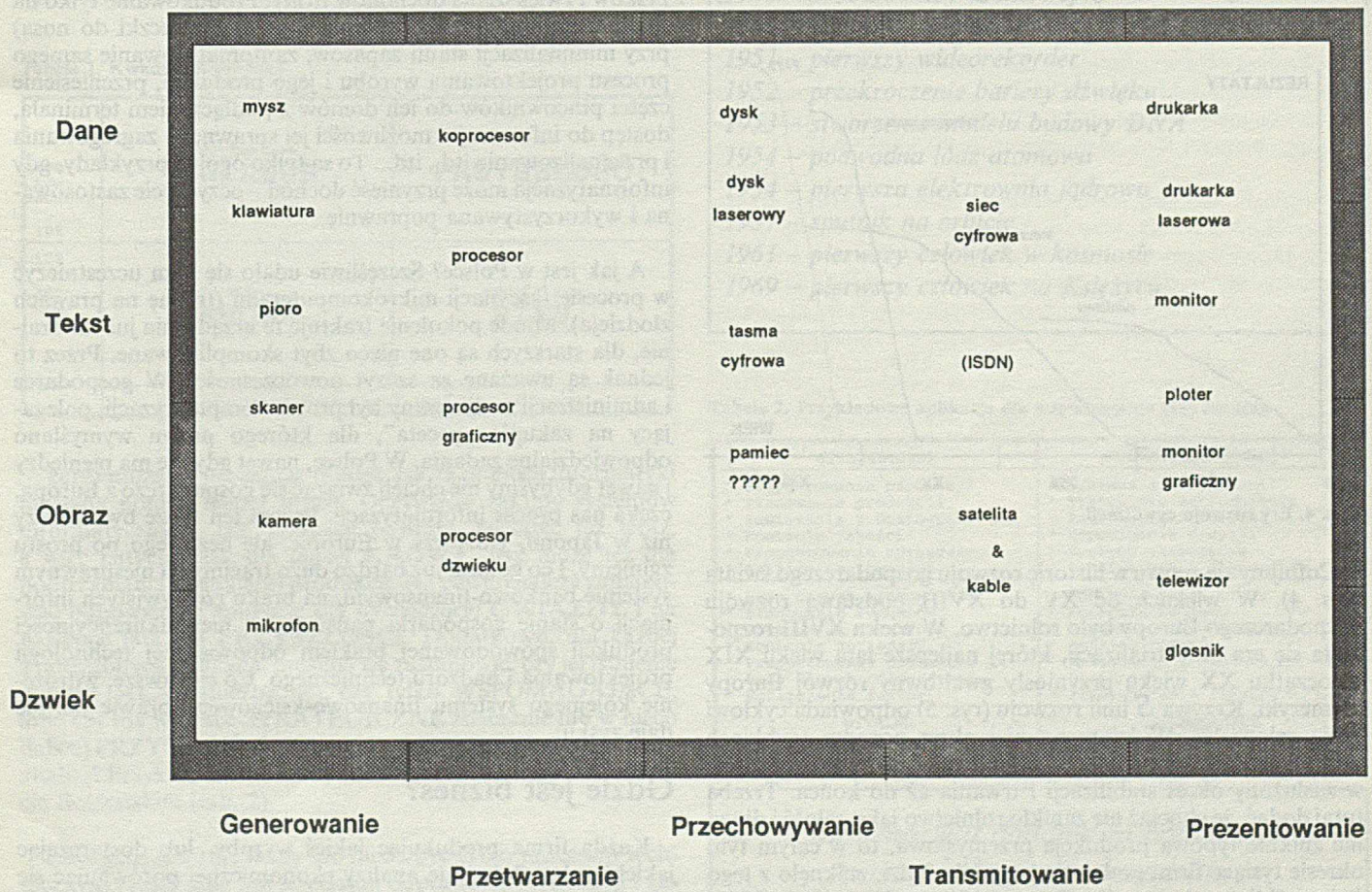
W kolejnych latach komputery zaczęły wchłaniać inne urządzenia techniczne: najpierw zaczęły przetwarzać teksty, a potem – prezentować na ekranach obraz (najpierw 2D, potem 3D i wreszcie animowany). Rozwijając sieci komputerowe spotykamy się z tym, że coraz więcej usług telekomunikacyjnych realizują komputery (rys. 3). A obecnie i w niedalekiej przyszłości wszystkie usługi będą realizowane przez system zwany informatycznym, gdyż komputer – a raczej procesor będzie tylko elementem tego systemu. Do generowania będzie wykorzystywana klawiatura zapewne inna niż używana dzisiaj: mysz, mikrofon, skaner i płytka do odczytu pisma ręcznego. Komputer będzie zajmował się przetwarzaniem wszystkich form przekształconych w jednolitą formę cyfrową, a następnie ich

[1] Targowski Andrzej: Informatyka kluczem do dobrobytu. PWN 1970

[2] *Informationalization* jest nowym słowem (na wzór słowa *industralization*) w języku amerykańskim (niektórzy wolą *infotize, infomate, informate, infotainment, infomediary, infotrate*) zbudowanym dla potrzeb opisu technologii przetwarzania informacji, któremu możemy spokojnie przypisać polskie słowo *informatyzacja*.



Rys. 2. Przykłady środków przekazu informacji



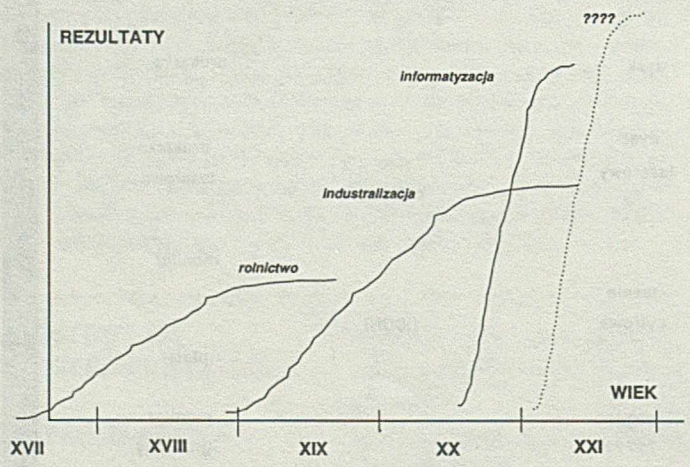
Rys. 3. Przyszłość środków przekazu informacji

mieszaniem. Do przechowywania będą wykorzystywane dyski laserowe i taśmy cyfrowe oraz inne rodzaje pamięci. Do transmisji zostanie użyta światłowodowa sieć ISDN, docierająca do każdego mieszkania, oraz ogólnoświatowa sieć satelitarna i lokalna radiowa do połączenia ze sobą wszystkich nadajników i odbiorników. Do prezentacji zostaną wykorzystane telewizory cyfrowe nowych typów i inne mocno zmodernizowane urządzenia audiowizyjne, jak na przykład wyświetlacze – wielkości kartki papieru.

To nie żart, ale nawet pójście do ubikacji będzie obcowaniem z informatyką, która w tym wypadku przekaże do centrum medycznego dane o naszym zdrowiu, a poza tym dla uniknięcia straty czasu zapewni nam łączność ze światem. Jeśli można telefonować z samochodu, to dlaczego nie z ubikacji. Czy to nie jest agresja informatyki w nasz intymny mały świat oraz w całe społeczeństwo?

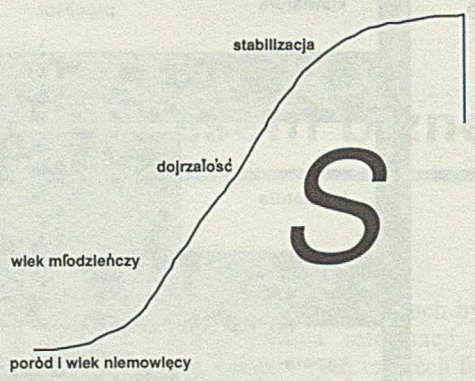
Informatyzacja to coś więcej niż komputeryzacja

Dlaczego świat zgadza się na taką agresję, chociaż trzeba przyznać, że wzorem robotników protestujących przeciwko maszynom parowym, wielu nie chce tego akceptować, bo nie rozumie zasad funkcjonowania i boi się skutków wyrzucenia poza nawias społeczny. Istnieje tylko jedyna przyczyna – ekonomia. Informatyka sama w sobie dostarcza tylko coraz mniejszych dochodów producentom sprzętu i oprogramowania, nieco większych projektantom systemów, a także konsultantom. Samo zakupienie komputera i systemu informatycznego nie przynosi kupującemu praktycznie żadnych korzyści finansowych, a tylko dużo kosztuje. W wielu instytucjach pracujący tam sprzęt jest już zamortyzowany tylko na papierze, a nie przyniósł żadnych dochodów oprócz splendoru posiadania nowoczesnej techniki. A jednak to tylko informatyka może być katalizatorem wzrostu gospodarczego.



Rys. 4. Ery rozwoju cywilizacji

Cofnijmy się znowu w historię rozwoju gospodarczego świata (rys. 4). W wiekach od XV do XVIII podstawą rozwoju gospodarczego Europy było rolnictwo. W wieku XVIII rozpoczęła się era industrializacji, której najlepsze lata wieku XIX i początku XX wieku przyniosły gwałtowny rozwój Europy i Ameryki. Krzywa S linii rozwoju (rys. 5) odpowiada cyklowi życia człowieka. Widzimy na niej okres porodu, szybkiego wzrostu, okres wchodzenia w dojrzałość, a następnie przejście w zasłużony okres stabilizacji i trwania aż do końca. Trzeba tutaj dodać, że chociaż nie zanikło rolnictwo jako całość i długo nie zniknie typowa produkcja przemysłowa, to w całym tym okresie tysiące firm, podobnie jak wielu ludzi, zniknęło z tego świata. Z listy miesięcznika *Fortune* z 500 istniejących w 1955 r. ponad 60% firm dzisiaj już nie istnieje. Żadna, nawet najlepiej zorganizowana firma nie może dzisiaj żyć w świadomości, że będzie istnieć wiecznie. Sprzedaż, likwidacja czy bankructwo jest wpisane w historię każdej firmy już w chwili jej powstania.



Rys. 5. Krzywa S

Pierwsze komputery powstały w latach 40. Początek ery informatycznej nastąpił w 1950 r. Przez wiele następnych lat wpływ informatyki na życie społeczeństwa i rozwój gospodarczy był niewielki. Budowano coraz kosztowniejsze maszyny, pisano coraz dłuższe programy, komputeryzowano banki i księgowość, a wreszcie powstały mikrokomputery, które weszły pod strzechy. Lata dziewięćdziesiąte są początkiem okresu dojrzałości informatyki i jej gwałtownego wpływu na rozwój gospodarczy świata, ale pojedynczo tylko tych firm, które zrozumieją, jak zastosowanie informatyki zamienić w zyski. Era informatyzacji ma trwać tylko 70 lat, tzn. do roku 2020 (to już za 28 lat!).

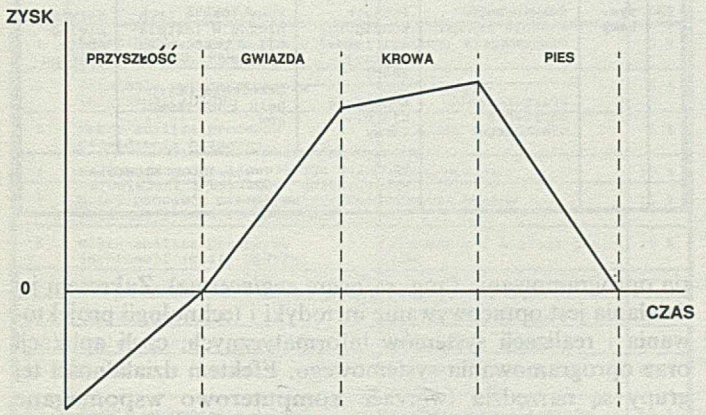
Powstanie sieci informatycznych i dostęp do systemu z każdego prawie miejsca pobytu jest właśnie tym kluczem do uzyskania zysków i zwiększenia dochodów firmy. Produkowanie tylko na zamówienie konkretnego klienta (nawet chusteczki do nosa) przy minimalizacji stanu zapasów, zautomatyzowanie samego procesu projektowania wyrobu i jego produkcji, przeniesienie części pracowników do ich domów z podłączeniem terminala, dostęp do informacji i możliwości jej sprawnego zagregowania i przeanalizowania itd., itd... To są tylko ogólne przykłady, gdy informatyzacja może przynieść dochód – oczywiście zastosowana i wykorzystywana poprawnie.

A jak jest w Polsce? Szczęśliwie udało się nam uczestniczyć w procesie fascynacji mikrokomputerami (trochę na prawach złodzieja). Młode pokolenie traktuje te urządzenia już naturalnie, dla starszych są one nieco zbyt skomplikowane. Przez to jednak są uważane za szczyt nowoczesności. W gospodarce i administracji realizowany był proces komputeryzacji, polegający na zakupie „peceta", dla którego potem wymyślano odpowiedzialne zadania. W Polsce, nawet gdy nie ma pieniędzy i nawet gdybyśmy nie chcieli związać się gospodarczo z Europą, czeka nas proces informatyzacji. Proces ten może być dłuższy niż w Japonii, USA czy w Europie, ale bez niego po prostu zginiemy. I co gorsza, już bardzo dużo tracimy na niesprawnym systemie bankowo-finansowym, na braku rzeczywistych informacji o stanie gospodarki państwa, na niekonkurencyjności produkcji spowodowanej brakiem odpowiedniej technologii projektowania i nadzoru technicznego. Co ciekawsze, wdrożenie kolejnego systemu finansowo-księgowego prawie już nie daje zysku.

Gdzie jest biznes?

Każda firma produkując jakieś wyroby, lub dostarczając jakichś usług dokonuje analizy ekonomicznej porównując się z konkurencją i obserwując swoje zyski oraz oczywiście starając

się myśleć o przyszłości. Obrazem takiej analizy może być Kwadrat Bostoński (tab.1) (opracowany przez *Boston Consulting Group* w latach 1960–1970) oraz jego rozwinięcie w postaci wykresu (rys. 6). Lewa dolna część kwadratu – *dojna* **KROWA** oznacza opanowanie danym produktem ponad 50% rynku przy stosunkowo niskim wskaźniku rentowności, czyli duży obrót – mały zysk jednostkowy. Prawa dolna część kwadratu – *zdychający* **PIES** oznacza upadającą wartość oferowanego produktu – duża konkurencja wypiera nas z rynku, a szanse wzrostu są niewielkie. Górna część kwadratu oznacza przyszłość. Po lewej stronie mamy *wschodzącą* **GWIAZDĘ**, czyli produkt będący „hitem" na rynku. Oczywiście każdy taki produkt staje się z czasem *dojną* **KROWĄ**, a potem *zdychającym* **PSEM**. Biznes jest w tym, aby stało się to jak najpóźniej, po zwrocie poniesionych nakładów. Prawa górna część kwadratu jest naszą **PRZYSZŁOŚCIĄ**. Jeżeli nie mamy tutaj nic do zaoferowania, to szanse przeżycia naszej firmy są małe.



Rys. 6. Zyski z produkcji *Cosia*

Tabela 1. Kwadrat Bostoński

Rentowność > 10%	*wschodząca* **GWIAZDA**	**PRZYSZŁOŚĆ**
10% < Rentowność	*dojna* **KROWA**	*zdychający* **PIES**
	pozycja na rynku >50%> pozycja na rynku	

Teraz możemy już zacząć myśleć, jak za pomocą informatyki możemy pomóc naszej firmie. Oczywiście zakupy systemów informatycznych, tzn. aplikacji powinny nastąpić już teraz, a więc natychmiast, dla *dojnej* **KROWY**, być może też dla podtrzymania upadającej produkcji. Na pewno należy zaplanować odpowiednie aplikacje dla *wschodzącej* **GWIAZDY** i na następne lata dla **PRZYSZŁOŚCI**. Dodatkowo, dla każdej z potrzebnych aplikacji, musimy się zastanowić, czy jej istnienie jest WITALNE (niezbędne), czy tylko WSPOMAGAJĄCE, albo czy ma wkrótce STRATEGICZNE znaczenie lub w nieco dalszej PRZYSZŁOŚCI. Proste, prawda, tylko jak uzyskać taką analizę? Przykład efektu takiej analizy jest pokazany w Kwadracie Bostońskim (tab. 2).

Informatyzowanie firmy nie może być tylko procesem technicznym znalezienia lub napisania odpowiedniej aplikacji i zainstalowania jej łącznie z komputerem. Wprowadzając informa-

Historia ery industrializacji

1733 – latające czółenko podwoiło prędkość tkania
1765 – maszyna dziewiarska zamiast 100 prządek
1784 – maszyna parowa Jamesa Watta
1784 – szybki rozwój przemysłu w Anglii i Płn. Stanach
1807 – pierwszy statek parowy
1825 – pierwsza na świecie linia kolejowa w Anglii
1839 – telegraf igiełkowy
1848 – przyspieszenie rozwoju przemysłu w Europie
1878 – organizacja badań – T. Edison
1880 – patent na żarówkę
1889 – przemysłowa wystawa światowa w Paryżu
1901 – fale radiowe pokonują Atlantyk
1903 – powstanie samolotu
1905 – seryjna produkcja samochodów w Detroit
1908 – Ford Model T – samochód powszechny
1913 – montaż taśmowy samochodu
1919 – przelot nad Atlantykiem
1919 – przekształcenie jądra atomu
1923 – 2 mln Fordów rocznie
1927 – popularność turystyki kolejowej
1935 – parowiec Normandie pokonał Atlantyk
1937 – otwarcie Golden Gate
1937 – pierwszy odbiornik telewizyjny
1940 – pierwszy komputer
1945 – pierwszy tranzystor
1945 – próba bomby atomowej
1950 – telewizor kolorowy
1951 – pierwszy wideorekorder
1952 – przekroczenie bariery dźwięku
1953 – stworzenie modelu budowy DNA
1954 – podwodna łódź atomowa
1954 – pierwsza elektrownia jądrowa
1957 – sputnik na orbicie
1961 – pierwszy człowiek w kosmosie
1969 – pierwszy człowiek na Księżycu

Tabela 2. Przykładowe aplikacje dla wspomagania fazy biznesu

STRATEGICZNE	PRZYSZŁOŚCIOWE
* Przygotowanie produkcji * Planowanie produkcji * Powiązania z dostawcami * Kontrola jakości * Prognozowanie sprzedaży * Analiza rentowności	* Wymiana z dostawcami * Planowanie zatrudnienia * Poczta elektroniczna * Wspomaganie decyzji * Systemy ekspertowe * Przetwarzanie dokumentów
* Baza danych pracowników * Planowanie remontów * Gospodarka magazynem * Kontrola sprzedaży * Projektowanie produktów * Baza danych półproduktów * Rozliczenia faktur	* Kontrola czasu pracy * Kontrola budżetu * Raportowanie wydatków * Analiza kosztów * Księgowość i finanse * Płatności * Przetwarzanie tekstów
WITALNE	WSPOMAGAJĄCE

Informatyka dodana do bałaganu tworzy bałagan do kwadratu.

tykę do firmy nie można popełnić błędów fabrykantów z początku XIX wieku. Systemy informatyczne tak bardzo wpływają na zmianę zachowań użytkowników, że bez zmian w świadomości pracowników nie można oczekiwać zakładanych efektów. Kto i w jaki sposób może dokonać odpowiedniej analizy organizacji zarządzania firmą oraz zespołem ludzi, a następnie zaproponować sensowne rozwiązanie porządkujące?

Realizacja systemu informatyzacyjnego

Zanim odpowiemy na te pytania, zbudujemy model życia systemu informatyzacyjnego, włączając doń wnioski z ostatnich rozważań.

SYSTEM INFORMATYZACYJNY
jest systemem obejmującym funkcjonowanie ekonomiczne i organizacyjne firmy odwzorowane w aplikacji systemu informatycznego.

W modelu cyklu życia takiego systemu są wiązane ze sobą rzeczy z czynnościami, które muszą być dla nich wykonane. W naszym modelu wyróżniamy dwie grupy rzeczy – BIZNES i INFORMACJE. W prowadzeniu BIZNESU firmy mamy cele (ang. *objectives*) jej rozwoju oraz **procesy** zachodzące wewnątrz i zewnątrz firmy. W realizacji przetwarzania INFORMACJI mamy **przepływy** (ang. *flows*) informacji w aplikacjach oraz systemy wykonujące zrealizowane aplikacje. Po stronie czynności wyróżniamy PLANOWANIE przedsięwzięć z opracowaniem **koncepcji** oraz **analizy**, a następnie PROJEKTOWANIE rozwiązania i REALIZACJĘ aplikacji systemu informatycznego z wykonaniem **implementacji i instalacji**. Nad prawidłowym przebiegiem tych prac czuwa ZARZĄDZANIE.

W modelu realizacji informatyzacji firmy znajdujemy wykaz czynności jakie powinny być wykonane, aby móc przygotować firmę, zaprojektować, zrealizować oraz eksploatować system informatyczny w firmie. Podany model, w którym można dyskutować stopień podziału na poszczególne fazy życia, przedstawia aktualną nowoczesną metodę informatyzacji firmy. Jest to jedyna prawidłowa droga – „OD ANALIZY CELÓW BIZNESU DO INSTALACJI KOMPUTERA", zamiast dotychczas najczęściej stosowanej komputeryzacji – „OD KUPNA KOMPUTERA DO POSZUKIWANIA DLA NIEGO PRACY".

Teraz możemy wstępnie wskazać metody wdrażania informatyki w organizację firmy. Przede wszystkim konieczne jest bardzo wczesne przeanalizowanie potrzeb i możliwości firmy. Co więcej, analizy te muszą być wykonane przez zewnętrzne w stosunku do firmy zespoły konsultantów przy ścisłej współpracy z dyrekcją. Równocześnie zespół konsultantów musi mieć odpowiednią wiedzę w zakresie ekonomii, menadżerstwa, profilu analizowanego biznesu oraz w możliwościach informatyki, czyli znajomości istniejących aplikacji, oprogramowania i sprzętu oraz narzędzi wspomagających.

Inżynieria informatyzacji

W ostatnich latach – w Polsce jeszcze dość nieśmiało – pojawiała się nowa zawodowa specjalność informatyczna – inżynieria oprogramowania (ang. *software engineering*). Zakresem jej działania jest opracowywanie metodyki i technologii projektowania i realizacji systemów informatycznych, czyli aplikacji oraz oprogramowania systemowego. Efektem działalności tej grupy są narzędzia tworzące **komputerowo wspomagane środowisko oprogramowania** (CASE – *Computer Aided Software Environment*). Obecnie jednak należy się zastanowić nad powstaniem nowej specjalności – inżynierii systemów (ang. *system engineering*) i równocześnie nieco zmienić znaczenie pojęcia CASE (... *Systems*...) na **komputerowo wspomagane środowisko systemów.**

Tabela 3. Model realizacji informatyzacji firmy

		PLANOWANIE		PROJEKTOWANIE	REALIZACJA		ZARZĄDZANIE
		Koncepcja	Analiza	Projekt	Implementacja	Instalacja	Operacja
BIZNES	**Cele**	Konsultacje w zarządzaniu firmą dla określenia jej celów Strategie wykorzystania informacji		Ewaluacja problemów Analiza potrzeb			
	Procesy	Makro-analiza procesów prowadzenia biznesu		Miary procesów przepływu informacji			
INFORMACJA	**Przepływy**	Mikro-analiza przepływu informacji w firmie		Zaprojektowanie lub przystosowanie aplikacji	Implementacja, testowanie i uruchomienie aplikacji		
		Konsultowanie wykorzystania gotowych aplikacji			Zarządzanie realizacją projektu		
	Systemy	Serwis wspomagania zakupu systemu		Projekt konfiguracji i integracji sieci	Konsultacje techniczne w instalacji i testowaniu systemu Szkolenia personelu i użytkowników		Serwis i pielęgnacja systemu
		Przygotowanie zapytania ofertowego		Konfiguracja platformy			
				Produkty sprzętowe i oprogramowania - wybór, zakup i dostawa i instalacja eksploatacyjna			

Co więcej, akceptując dotychczasowe rozważania o roli informatyzacji, warto już powołać i określić jej zadania, nową specjalność – inżynierię informatyzacji (ang. *informationalization engineering*). Zadaniem tych specjalistów byłaby integracja systemu informatyzacyjnego przez zaprojektowanie i zrealizowanie wszystkich faz prowadzących do uzyskania gotowego systemu, złożonego z aplikacji przygotowanych na podstawie szczegółowych, uzasadnionych ekonomicznie, analiz celów i potrzeb firmy. Oczywiście tacy specjaliści już istnieją (jest ich co prawda niewielu) oraz znane jest już wiele prac teoretycznych i metodycznych, jak zabrać się do tego dość złożonego problemu.

Obecnie elementy fazy PLANOWANIA są realizowane głównie przez konsultacje i sesje analityczne, zorganizowane według precyzyjnie przygotowanych scenariuszy. Stosowane są tutaj analizy mocy i słabości oraz możliwości i ograniczeń (SWOT – *Strengthes, Weaknesses, Opportunities, Threats*) funkcjonowania danej jednostki biznesu. Wykorzystuje się również metody aktywnych dyskusji oraz burzy mózgów. Często efektem takich sesji są proste w formie i czytelne dla uczestników plansze i schematy. Wartość tego typu konsultacji jest zależna od efektywnego udziału i zrozumienia problemu przez dyrekcję firmy[3].

[3] Koszt konsultacji kształtuje się w granicach od 1500 do 4000 dolarów za dzień pracy konsultanta, bez kosztów jego podróży i zakwaterowania, oraz samych kosztów organizacji szkolenia.

Narzędzie wspomagania informatyzacji

Inżynieria informatyzacji potrzebuje również wspomagania komputerem, czyli specjalnych narzędzi do analizy celów i potrzeb biznesu, a także możliwości wykorzystania informatyki dla zwiększenia efektywności finansowej firmy. Obecnie możemy jedynie odnotować pierwsze próby tworzenia takiego oprogramowania opartego na systemach ekspertowych. Większość tych narzędzi będzie początkowo wspomagać zbieranie i porządkowanie zbieranych informacji. Z czasem możliwe będzie opracowanie standardowych analiz dla typowych firm i działalności. Rozprzestrzenienie się systemów komputerowych z nowoczesnymi terminalami z dostępem do zintegrowanych systemów informacyjnych pozwoli na łatwiejszą analizę porównawczą ekonomicznego stanu firmy.

Teraz możemy oszacować (tab. 4) stopień istnienia, w stosunku do zapotrzebowania, narzędzi wspomagania dla inżynierii informatyzacji, w tym również dla inżynierii systemów i oprogramowania (jest to bardzo szacunkowa, subiektywna ocena autora). Wnioski wynikające z analizy tej tabeli, po pierwsze poddaję pod dyskusję Czytelników z oczekiwaniem na ich uzupełnienie, a po drugie – bez względu na wynik tej dyskusji – możemy podać jeden wniosek bezdyskusyjny zamieszczony niżej w ramce.

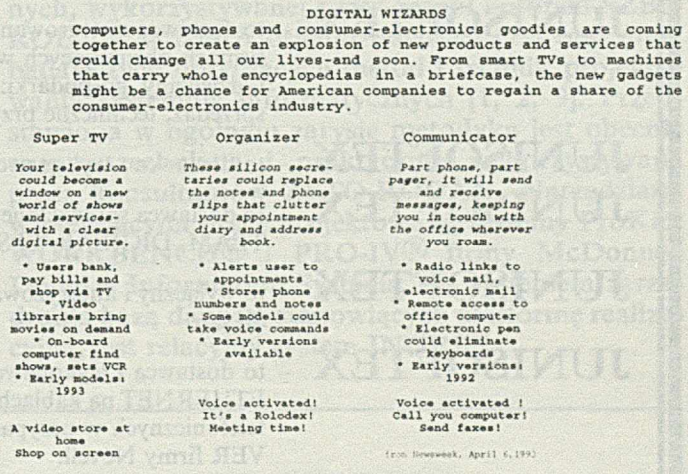
Tabela 4. Wspomaganie inżynierii informatyzacji

	ZADANIA	WSPOMAGANIE	CASE
1	Konsultacje w zarządzaniu firmą (TOP-MAPPING, RAMS)	systemy ekspertowe	5 %
2	Strategie wykorzystywania informacji (DISS)	bazy danych	5 %
3	Makro-analiza procesów prowadzenia biznesu	programy graficzne	5 %
4	Ewaluacja problemów	dokumentacja	10 %
5	Miary procesów przepływu informacji (met.YOURDONA,etc)	analiza grafów	15 %
6	Mikro-analiza przepływu informacji (grafy GANTTA)	rysowanie i analiza grafów	20 %
7	Konsultowanie aplikacji	bazy danych aplikacji	10 %
8	Projekt aplikacji	projektowanie, słowniki	50 %
9	Implementacja i testowanie	generatory, debugery	90 %
10	Zarządzanie realizacją projektu	systemy planowania i nadzoru zadań	80 %
11	Serwis wspomagania zakupu systemu	systemy analiz kosztów	60 %
12	Przygotowanie zapytania ofertowego	budowanie dokumentacji	70 %
13	Projekt konfiguracji	systemy projektowania sieci	60 %
14	Konfiguracja platformy	systemy weryfikacji konfiguracji	30 %
15	Produkty sprzętowe i programowe	"problem dostawcy"	
16	Konsultacje techniczne	systemy - multimedia	20 %
17	Szkolenia	systemy edukacyjne	20 %
18	Instalacja	"problem dostawcy"	
19	Serwis produktów	"problem dostawcy"	
20	Analiza realizacji systemu informatyzacyjnego	systemy gromadzenia dokumentacji	10 %

W inżynierii informatyzacji jest jeszcze bardzo dużo do zrobienia dla uzyskania efektywnych narzędzi wspomagających analizę, projektowanie i realizację, a następnie pielęgnację złożonych systemów informatyzacyjnych.

Zamiast podsumowania

Zaprezentowanego materiału nie można podsumować. Jest to bowiem jeszcze nie uporządkowane spojrzenie na to co nas czeka, a więc jest to w pewnym stopniu futurologia. Jaką mamy gwarancję, że spełnią się założenia totalnej informatyzacji



DIGITAL WIZARDS

Computers, phones and consumer-electronics goodies are coming together to create an explosion of new products and services that could change all our lives-and soon. From smart TVs to machines that carry whole encyclopedias in a briefcase, the new gadgets might be a chance for American companies to regain a share of the consumer-electronics industry.

Super TV

Your television could become a window on a new world of shows and services- with a clear digital picture.

* Users bank, pay bills and shop via TV
* Video libraries bring movies on demand
* On-board computer find shows, sets VCR
* Early models: 1993

A video store at home
Shop on screen

Organizer

These silicon secretaries could replace the notes and phone slips that clutter your appointment diary and address book.

* Alerts user to appointments
* Stores phone numbers and notes
* Some models could take voice commands
* Early versions available

Voice activated!
It's a Rolodex!
Meeting time!

Communicator

Part phone, part pager, it will send and receive messages, keeping you in touch with the office wherever you roam.

* Radio links to voice mail, electronic mail
* Remote access to office computer
* Electronic pen could eliminate keyboards
* Early versions: 1992

Voice activated !
Call you computer!
Send faxes!

(from Newsweek, April 6,1992)

gospodarki i społeczeństwa. Jest to bardzo bliska przyszłość i chyba nam grozi. A co nas, lub nasze dzieci czeka po roku 2020? Tutaj jest już trochę trudniej stwierdzić, jaka era nastąpi potem. Ja w każdym razie mogę opowiedzieć następującą historyjkę.

Kiedyś mój czternastoletni syn stwierdził, że nie chce zajmować się informatyką. Na pytanie – *a czym chciałbyś się zajmować?* – odpowiedział – *chcę się zajmować biologią*. Wtedy wyjaśniłem mu, że to będzie właśnie przyszłość nowoczesnego świata i komputery będą bioniczne. Nieco się załamał tą perspektywą, że jednak będzie to bioputer, albo coś podobnego.

Wskazówki dla Autorów

Nadsyłane artykuły nie mogą być publikowane lub przeznaczone do opublikowania w innych czasopismach.

Materiał oprócz tekstu zasadniczego powinien zawierać:

- ☆ krótki życiorys zawodowy Autora i jego zdjęcie,
- ☆ materiał ilustracyjny (rysunki, zdjęcia czarno-białe, wydruki) dołączony do artykułu (nie wklejać materiału w tekst),
- ☆ podpisy pod ilustracje.

Na osobnej stronie prosimy podać: tytuł naukowy, nazwisko i imię, nazwę zakładu pracy, numer telefonu służbowego oraz informację, jaką drogą przekazać honorarium – kasa Wydawnictwa, poczta, konto.

W związku z naliczaniem podatku dochodowego prosimy także podać:

- ☆ oprócz nazwiska i imienia – także drugie imię,
- ☆ imiona rodziców,
- ☆ datę i miejsce urodzenia,
- ☆ numer identyfikacyjny PESEL,
- ☆ miejsce zameldowania,
- ☆ adres Urzędu Skarbowego właściwego dla miejsca zamieszkania Autora.

Tekst artykułu prosimy dostarczyć w formie maszynopisu lub wydruku komputerowego, pisanego jednostronnie, z podwójnym odstępem (30 wierszy na stronie i 60 znaków w wierszu).

Rysunki powinny być czytelne i mogą być wykonane ołówkiem.

Autor opublikowanego artykułu otrzymuje honorarium i bezpłatny egzemplarz okazowy.

Materiałów nie zamówionych redakcja nie zwraca.

JUNISOFTEX Sp. z o.o.

44-100 GLIWICE ul. Konstytucji 11,
tel.-faks 31-75-10, 31-90-81 do 88 w. 250, 272, 282, teleks 036233

JUNISOFTEX – to firma z tradycjami i najdłuższymi doświadczeniami w eksploatacji wielodostępnych systemów komputerowych Novell w kraju.

JUNISOFTEX – to najstarszy w kraju wykonawca własnych wielodostępnych systemów pracujących w sieci NetWare firmy Novell.

JUNISOFTEX – to autoryzowany reseller amerykańskiej firmy Novell.

JUNISOFTEX – to dostawca zintegrowanych systemów komputerowych działających w kilkudziesięciu firmach obejmujących w każdym przedsiębiorstwie, niezależnie od formy własności i dziedziny gospodarki: finanse, majątek obrotowy, majątek trwały i nietrwały, sprzedaż, techniczne przygotowanie produkcji oraz kadry-płace.

JUNISOFTEX – to dostawca i wykonawca autoryzowanych sieci lokalnych firmy Novell.

JUNISOFTEX – to dostawca sprawdzonego sprzętu komputerowego renomowanych firm ALR, IBM, TEAM, DIGILAB, EPSON, MANNESMANN TALLY.

JUNISOFTEX – to wyłączny i autoryzowany dystrybutor komputerów i terminali firmy DIGILAB na Śląsku.

JUNISOFTEX – to dostawca i wykonawca okablowania lokalnych sieci komputerowych ARCNET, ETHERNET na kablach zwykłych i światłowodowych oraz na dwużyłowych kablach telefonicznych w przypadku stosowania oprogramowania NetWare ACCESS SERVER firmy Novell.

JUNISOFTEX – to nauczyciel, który chętnie podzieli się swoją wiedzą na organizowanych kursach w swojej szkole informatycznej – w tym nauczy Cię: podstaw informatyki, obsługi sprzętu komputerowego, eksploatacji własnych systemów informatycznych, edytorów tekstu: CHiWRITER, WordPerfect, arkuszy kalkulacyjnych QPRO v. 3.0, LOTUS 1-2-3 itp.

JUNISOFTEX – to doradca w zakresie komputerowych metod organizacji i eksploatacji systemów komputerowych.

JUNISOFTEX – to strażnik postępu i nowoczesności w Twojej firmie.

JUNISOFTEX – to gwarancja niezawodności, rzetelności i terminowości.

JUNISOFTEX – to Twój doradca i partner, któremu możesz zaufać, który Cię nigdy nie zawiedzie.

JUNISOFTEX – to partner, który Cię wysłucha i zawsze pomoże podjąć dobrą decyzję.

Jeżeli chcesz

- ☆ zastosować najnowocześniejsze systemy komputerowe oraz lokalne sieci komputerowe,
- ☆ zreorganizować, usprawnić i szybko skomputeryzować swoje przedsiębiorstwo,
- ☆ lepiej wykorzystać już istniejące w Twoim przedsiębiorstwie komputery,
- ☆ wymienić swoje niefortunnie zakupione oprogramowanie,
- ☆ pozbyć się problemów eksploatacyjnych, upadających systemów, nie działających komputerów,
- ☆ przeszkolić załogę swojego przedsiębiorstwa,
- ☆ dobrze zainwestować swoje pieniądze i podnieść rangę swojego przedsiębiorstwa,
- ☆ w przyszłości mieć dostęp do krajowej i światowej sieci komputerowej,

TO ZGŁOŚ SIĘ DO NAS!

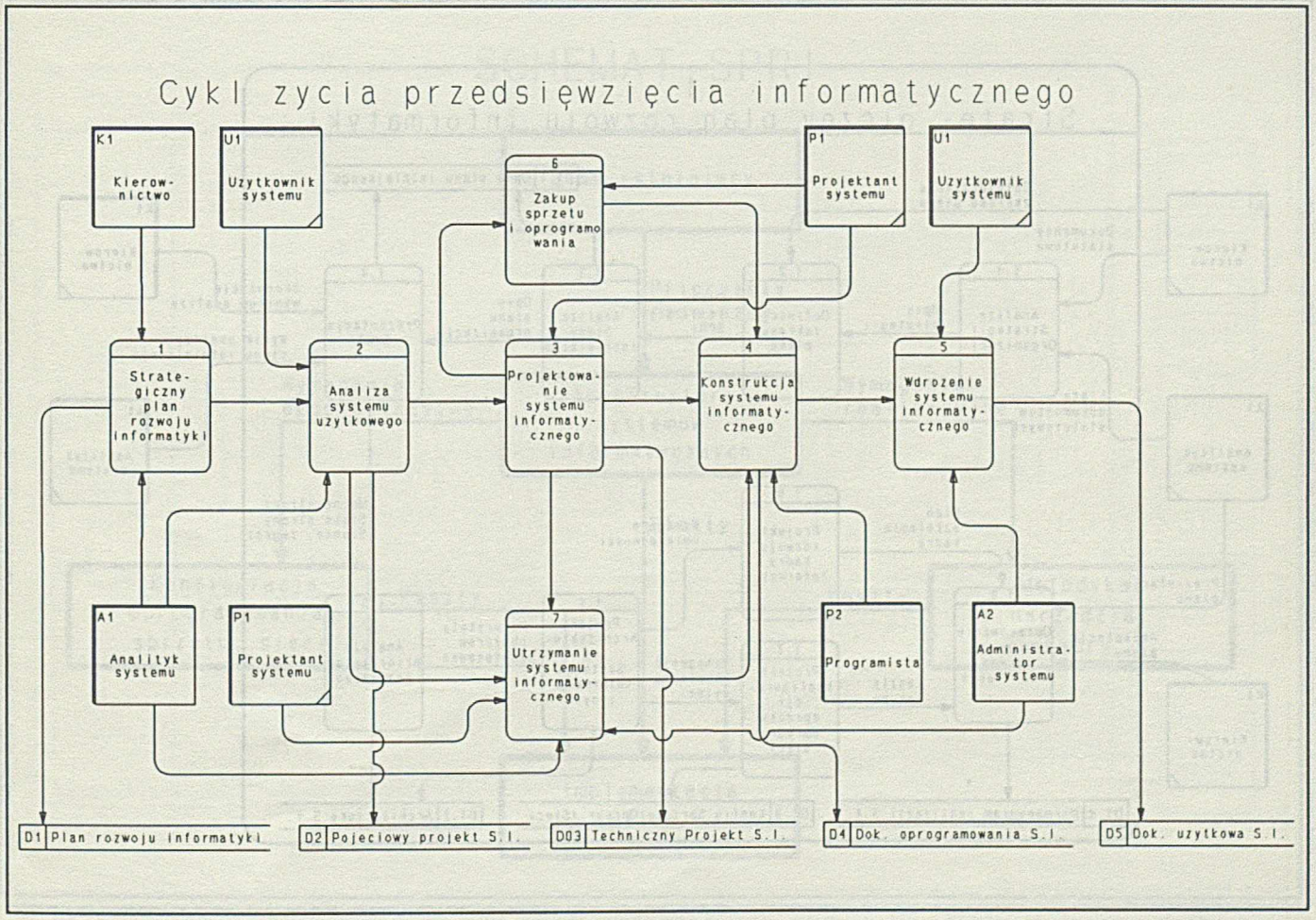
0/17/91

TERESA PIÓRKOWSKA
DARIUSZ PUTERNICKI
WITOLD STANISZKIS
ZETO-RODAN
Warszawa

Inżynieria systemów informatycznych od strategii do realizacji

W artykule przedstawiono zarys metodyki projektowania i realizacji systemów informatycznych, stosowanej i rozpowszechnianej przez firmę ZETO-RODAN. Poza opisem metodyki, przedstawiono podstawowe systemy oprogramowania narzędziowego (ProKit*WORKBENCH®, PRO-IV®) wykorzystywane w pracach projektowo-programowych oraz rozpowszechniane w Polsce przez ZETO-RODAN.

Prezentowana metodyka stanowi ewolucję praktycznych doświadczeń w zakresie projektowania systemów użytkowych opartych na wspólnej bazie danych, wykorzystywanej przez zespół autorski SZBD RODAN w latach osiemdziesiątych [15] oraz prac badawczych dotyczących metod i technik projektowania systemów informatycznych [1, 2, 9]. Przedstawiona w ogólnym zarysie metodyka jest obecnie rozpowszechniana i praktycznie wykorzystywana przez konsultantów ZETO-RODAN. Narzędziami wspierającymi prace projektowe są systemy ProKit* WORKBENCH® i PRO-IV® firmy McDonnell Douglas Information Systems, zaś systemem zarządzania bazą danych, stanowiących platformę realizacyjną, jest relacyjny system INGRES.



Rys. 1. Zakres metodyki projektowania i realizacji systemów informatycznych

Schemat cyklu życia przedsięwzięcia informatycznego przedstawiono w formie schematu przepływu danych na rys. 1. Wykorzystywana w naszej metodyce notacja schematów przepływu danych jest zgodna z propozycją Chrisa Gane'a i Trisha Sarsona [4]. Wszystkie schematy przepływu danych przedstawione w tej publikacji zostały wykonane przy użyciu edytora graficznego systemu ProKit*WORKBENCH®.

Przedstawiony cykl życia przedsięwzięcia informatycznego obejmuje wszystkie podstawowe fazy cyklu życia systemu informatycznego, zgodnie z analizą wykonaną przez zespół IFIP pod kierownictwem Billa Olle'a [10], poczynając od strategicznego planu rozwoju informatyki, a kończąc na utrzymaniu systemu informatycznego.

W dalszych częściach artykułu przedstawiono trzy podstawowe fazy cyklu życia przedsięwzięcia informatycznego, a mianowicie strategiczny plan rozwoju informatyki oraz analizę i projekt techniczny systemu użytkowego, a także systemy oprogramowania narzędziowego ProKit*WORKBENCH® oraz PRO-IV®.

Metodyka ZETO-RODAN jest rozpowszechniana w formie szkoleń i usług wdrożeniowych u użytkowników produktów oprogramowania narzędziowego (ProKit*WORKBENCH®, PRO-IV®, INGRES). Licencja pozwalająca na wykorzystanie metodyki jest przyznawana nieodpłatnie wszystkim użytkownikom produktów programowych rozpowszechnianych przez naszą firmę.

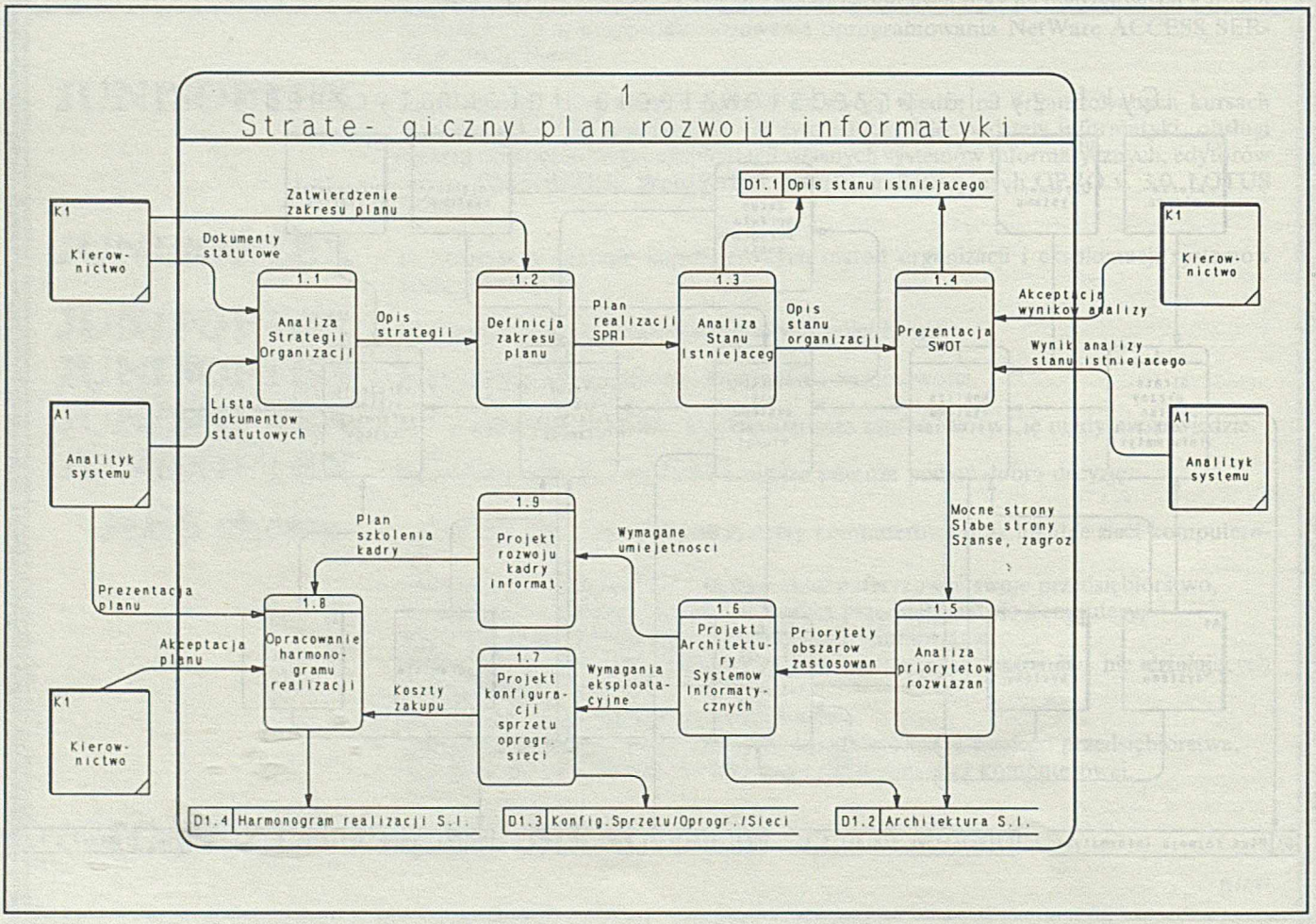
Strategiczny plan rozwoju informatyki

Strategiczny Plan Rozwoju Informatyki (SPRI) jest dokumentem decyzyjnym wyznaczającym kierunki rozwoju zastosowań informatyki w okresie przyjętym jako horyzont czasowy planu. Typowym horyzontem czasowym SPRI jest okres 4–5 lat. Istotą tego dokumentu jest konfrontacja rzeczywistych problemów i ograniczeń funkcjonowania organizacji z możliwościami wynikającymi z bieżącego stanu rozwoju informatyki. Proponowane rozwiązania obejmują architekturę sprzętu komputerowego i oprogramowania podstawowego, jak również zakres funkcji i model informacyjny proponowanych systemów informatycznych.

Proces realizacji strategicznego planu rozwoju informatyki przedstawiono w formie schematu przepływu danych na rys. 2. Strukturę dokumentacji SPRI przedstawiono na rys. 3.

Podstawowym elementem planu strategicznego jest preferencyjna lista systemów informatycznych stanowiących rozwiązanie istniejących ograniczeń funkcjonowania organizacji. Lista proponowanych rozwiązań, zaakceptowana przez kadrę kierowniczą, stanowi podstawę dalszych prac nad planem.

Punktem wyjściowym części informatycznej SPRI jest definicja architektury systemów informatycznych przewidzianych do realizacji w okresie objętym horyzontem czasowym planu. Istotnym elementem prezentacji proponowanej architektury



Rys. 2. Proces realizacji strategicznego planu rozwoju informatyki

systemów informatycznych jest tablica krzyżowa ilustrująca korelację elementów tej architektury z ograniczeniami funkcjonowania organizacji, zidentyfikowanymi w trakcie analizy stanu istniejącego.

Dalsze części planu obejmują uwarunkowania technologiczne realizacji proponowanej architektury systemów informatycznych, przedstawione w formie architektury sprzętu komputerowego, oprogramowania oraz sieci transmisji danych, a także w formie opisu metodyki i narzędzi realizacji proponowanych systemów informatycznych oraz wymagań dotyczących kadry informatycznej.

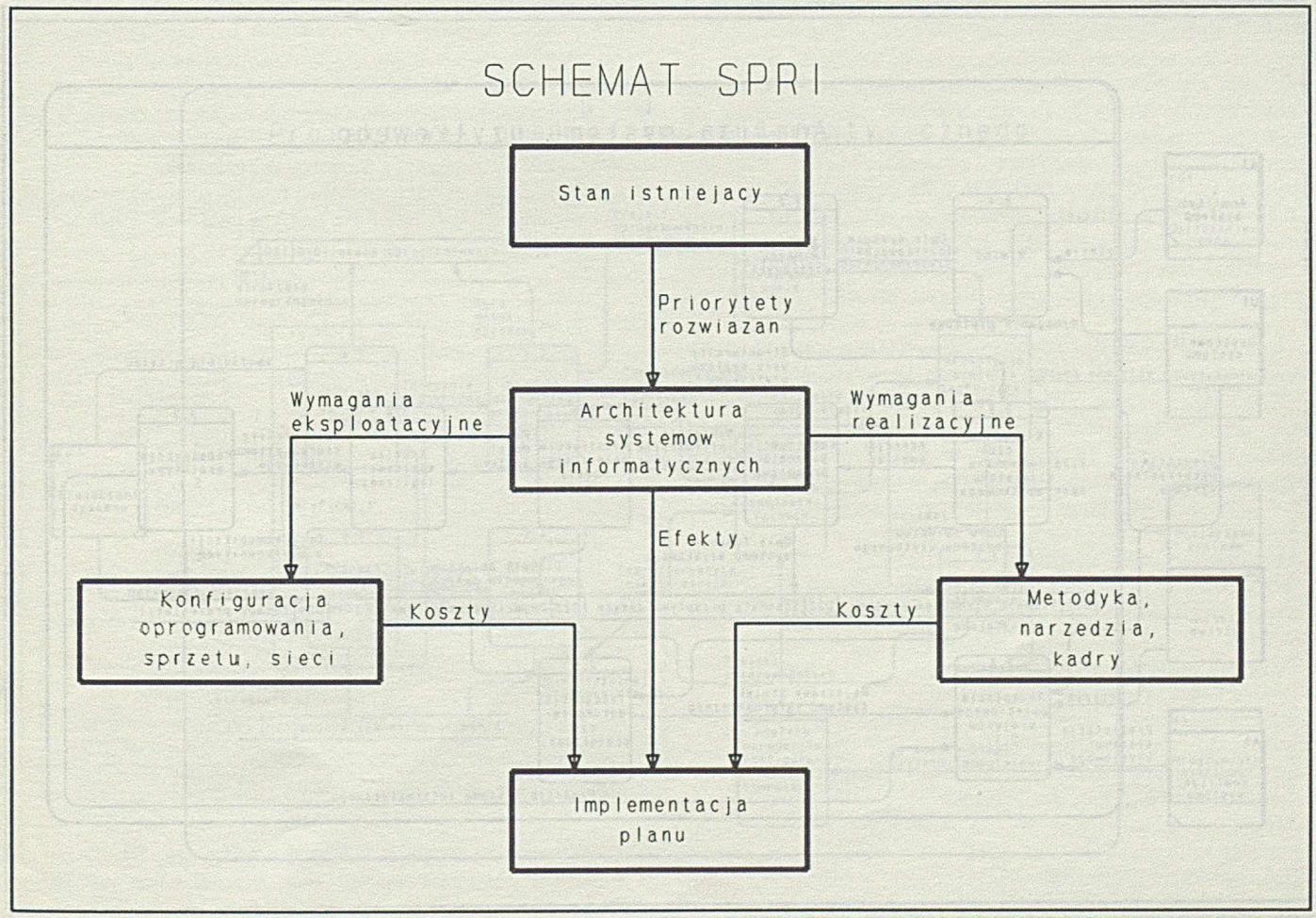
Podsumowaniem jest plan realizacji obejmujący podstawowe fazy przewidzianych prac nad systemami informatycznymi. Specyfika prac informatycznych nie pozwala na dokładne określenie pracochłonności wszystkich kolejnych faz realizacji systemu informatycznego. Proponujemy metodę kolejnych przybliżeń oceny kosztów realizacji, przyjmującą zasadę dokładnego określania pracochłonności następnej fazy, oraz przybliżone szacowanie pracochłonności dalszych etapów pracy. Ostateczna akceptacja proponowanego strategicznego planu rozwoju informatyki odbywa się na podstawie analizy planu finansowego, obejmującego cały okres realizacji planu. Plan finansowy powinien określać wszystkie podstawowe elementy kosztów realizacji oraz analizę opłacalności proponowanych przedsięwzięć.

Analiza systemu użytkowego

Celem analizy systemu użytkowego jest opracowanie pojęciowego projektu systemu pozwalającego na pełną prezentację wszystkich aspektów użytkowych systemu informatycznego, jakie wynikają bezpośrednio z wymagań użytkowników. Podstawowymi elementami projektu pojęciowego są: słownik danych, schematy przepływu danych, pojęciowy model danych oraz prototyp przedstawiający zasady prezentacji danych. Istotnym elementem dokumentacji projektu pojęciowego jest również analiza wymagań eksploatacyjnych, obejmująca takie elementy, jak definicja ograniczeń czasowych funkcji użytkowych (czas odpowiedzi, tryb przetwarzania) oraz przewidywana zajętość pamięci zewnętrznej przez podstawowe zbiory danych. Proces analizy przedstawiono w formie schematu przepływu danych na rys. 4.

Słownik danych obejmuje wszystkie podstawowe elementy projektu pojęciowego (atrybuty, obiekty, funkcje użytkowe), identyfikowane jednoznacznie przez nazwy elementów słownika i opatrzone definicjami. Słownik danych systemu użytkowego stanowi podstawowy wynik procesu klasyfikacji elementów systemu.

Konstrukcja schematów przepływu danych prezentujących wynik analizy funkcji systemu użytkowego (aspekt dynamiczny) oraz pojęciowego modelu danych, stanowiącego wynik analizy danych (aspekt statyczny), jest procesem iteracyjnym. Poszczególne iteracje obu typów analiz prowadzą do kolejnych



Rys. 3. Struktura dokumentu strategicznego planu rozwoju informatyki

uściśleń reprezentacji dynamicznych i statycznych aspektów systemu informatycznego.

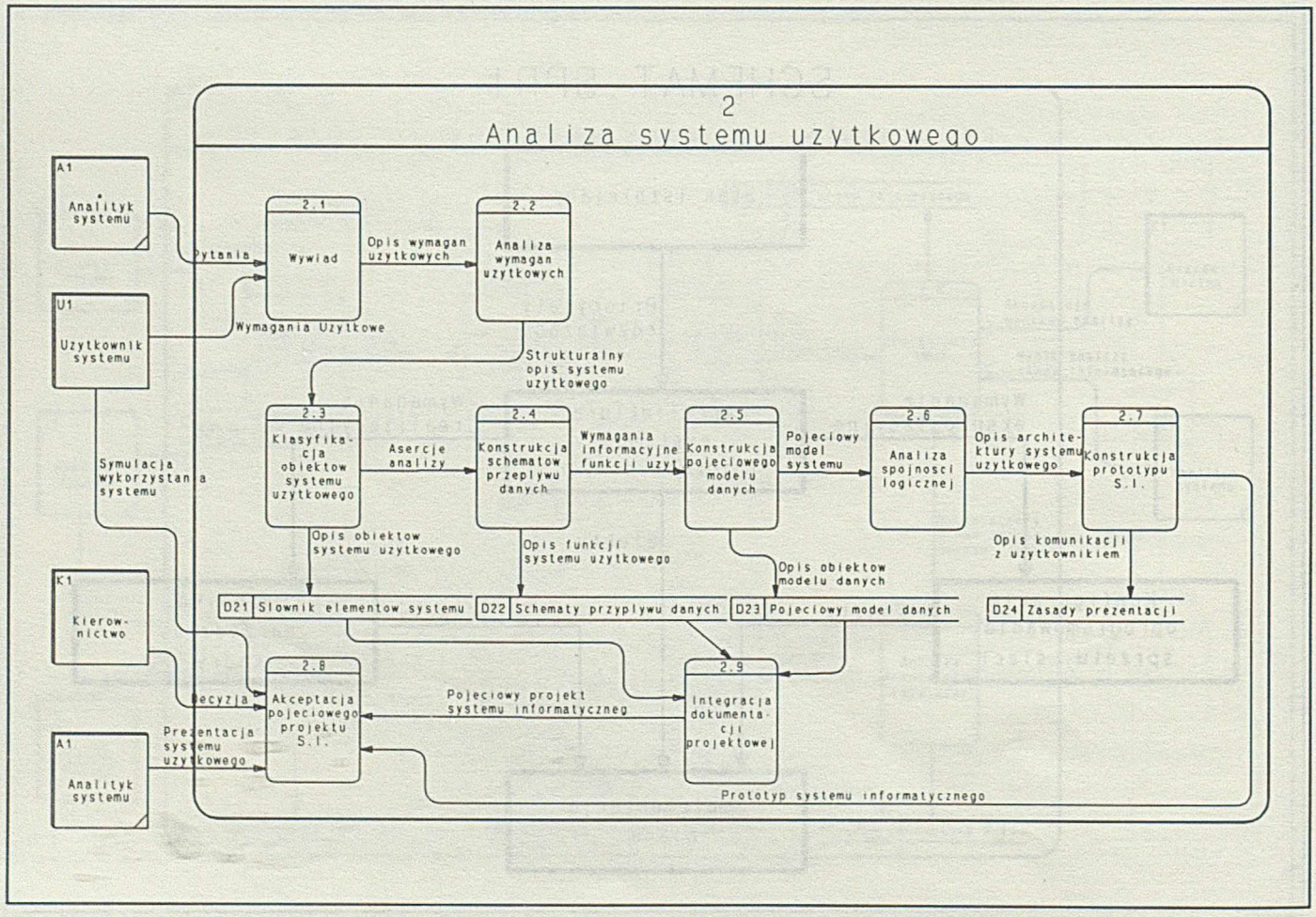
Ważnym etapem analizy systemu użytkowego jest analiza spójności logicznej pojęciowego modelu systemu. Jest ona realizowana automatycznie przez funkcje poziomego i pionowego bilansowania schematów przepływu danych zapisanych w encyklopedii (ang. *repository*) systemu ProKit*WORKBENCH®. Analiza pozwala na identyfikację wszystkich błędów formalnych w specyfikacji procesów oraz strumieni i zbiorów danych. Dodatkowym elementem analizy spójności logicznej jest konfrontacja asercji, wynikających z analizy wymagań użytkowych, z opisem pojęciowego modelu danych, obejmującym pojęciowy schemat danych oraz definicję reguł spójności logicznej. Dopuszczalne notacje graficzne pojęciowego modelu danych obejmują schematy struktur danych, schematy obiekt-powiązanie Chen'a oraz pojęciowe schematy relacyjne.

Istotnym elementem projektu pojęciowego jest prototyp interfejsu użytkownika, pozwalający na pełną prezentację ekranów proponowanego systemu, nawigacji po drzewie funkcji użytkowych systemu oraz diagnostyki i funkcji *help*. Konstrukcja prototypu w ramach projektu pojęciowego jest podstawowym warunkiem uzyskania świadomej akceptacji użytkownika, co z kolei prowadzi do znacznego uproszczenia fazy wdrażania systemu użytkowego. Prototyp jest realizowany całkowicie przy wykorzystaniu odpowiednich funkcji systemu ProKit*WORKBENCH®.

Projekt techniczny systemu użytkowego

Projekt techniczny systemu użytkowego jest całkowicie ukierunkowany na konkretne środowisko eksploatacyjne (system operacyjny, system ZBD, monitor TP), a podstawowym ograniczeniem tej fazy projektowej jest uzyskanie wymaganych parametrów eksploatacyjnych systemu. Przyjęta w naszej metodyce zasada jednokrotnego definiowania poszczególnych aspektów systemu użytkowego, przy zastosowaniu ścisłego podziału na poziomy dokumentacji projektowej, wymaga automatycznego przenoszenia elementów projektu pojęciowego do opisu projektu technicznego. Zastosowanie systemu ProKit*WORKBENCH® pozwala na automatyczne generowanie logicznego schematu relacyjnej bazy danych (standard SQL), opisów ekranów użytkowych, definicji struktury oprogramowania i struktury dialogów (PRO-IV®, CSP/AD, TELON, TRANSFORM) oraz deklaracji struktur danych w podstawowych językach trzeciej generacji. W tej sytuacji decyzje projektowe podejmowane w tej fazie prac dotyczą wyłącznie aspektów determinujących parametry eksploatacyjne projektowanego systemu. Zakres parametrów eksploatacyjnych objętych typowym projektem technicznym przedstawiono w [13].

Istotnym elementem projektu technicznego jest przewidywanie parametrów eksploatacyjnych, wynikających z decyzji projektowych podjętych w ramach rozpatrywanego wariantu projektu. Oczywistym warunkiem akceptacji tego wariantu jest spełnienie wymagań eksploatacyjnych ustalonych w ramach analizy systemu użytkowego. Przewidywanie wartości paramet-



Rys. 4. Proces analizy systemu użytkowego

rów czasowych, takich jak średni czas odpowiedzi poszczególnych klas funkcji użytkowych czy przepustowość systemu, wymaga oceny obciążeń konfiguracji systemu komputerowego, generowanych przez oprogramowanie użytkowe (liczba operacji we-wy, liczba instrukcji jednostki centralnej). Ocena powyższych parametrów wymaga realizacji dwupoziomowego modelu analitycznego, obejmującego analizę obciążeń (równania kosztów) oraz analizę funkcjonowania systemu komputerowego (modele masowej obsługi). Opis stosowanych w naszej metodyce technik i narzędzi programowych, obejmujących hierarchiczne modele oceny parametrów eksploatacyjnych, przedstawiono w [1, 11].

Ważnym elementem projektu technicznego jest plan realizacji oprogramowania użytkowego obejmujący wszystkie podstawowe fazy cyklu życia tego etapu prac nad systemem informatycznym. Warunkiem konstrukcji realistycznych planów prac jest ocena pracochłonności realizacji poszczególnych elementów architektury systemu. Wytyczne dotyczące oszacowania pracochłonności oraz organizacji prac programowych są przedstawione w sposób wyczerpujący w [5]. Wszystkie proponowane w tej pracy podstawowe elementy zostały wprowadzone do metodyki ZETO-RODAN.

Proces akceptacji projektu technicznego obejmuje takie elementy, jak ocena parametrów eksploatacyjnych systemu użytkowego, ocena planu realizacji oprogramowania oraz ustalenie zasad kontroli jakości oprogramowania. Proces projektowania systemu użytkowego przedstawiono w formie schematu przepływu danych na rys. 5.

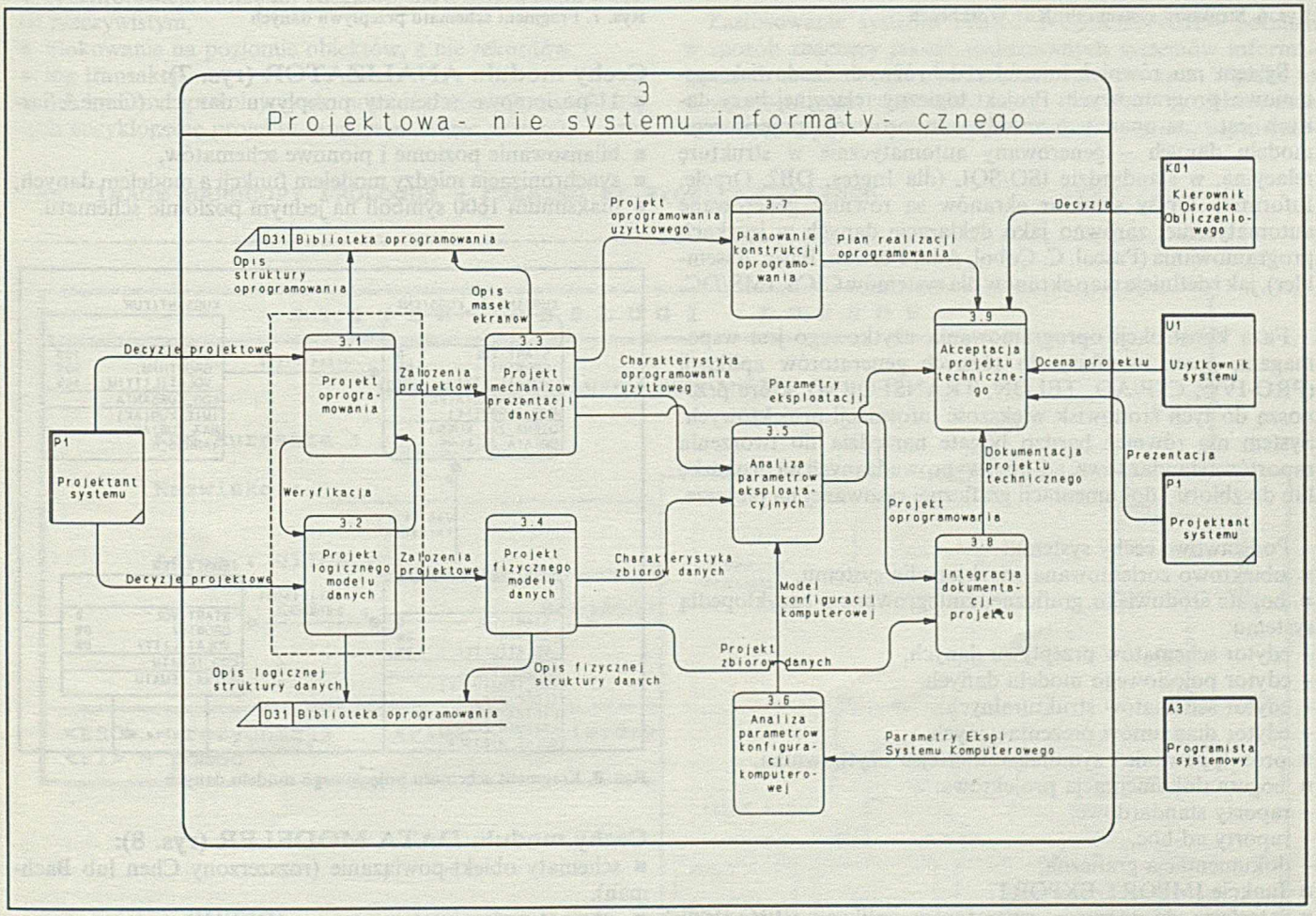
ProKit*WORKBENCH®

System **ProKit*WORKBENCH®** jest zaawansowanym, zintegrowanym narzędziem upper-CASE wspomagającym analityka i projektanta w trakcie całego cyklu życiowego systemu informatycznego – od planowania strategicznego poprzez analizę i projektowanie aż do utrzymania tego systemu.

Architekturę systemu ProKit*WORKBENCH® przedstawiono na rys. 6. Jest ona oparta na koncepcji stacji roboczej projektanta. Podstawowym elementem systemu jest obiektowo zorientowana encyklopedia, zawierająca pełną dokumentację projektową. Encyklopedia ta może być utrzymywana w ramach jednej stacji roboczej lub może być wspólną dla całego zespołu projektowego, dostępną w ramach sieci lokalnej.

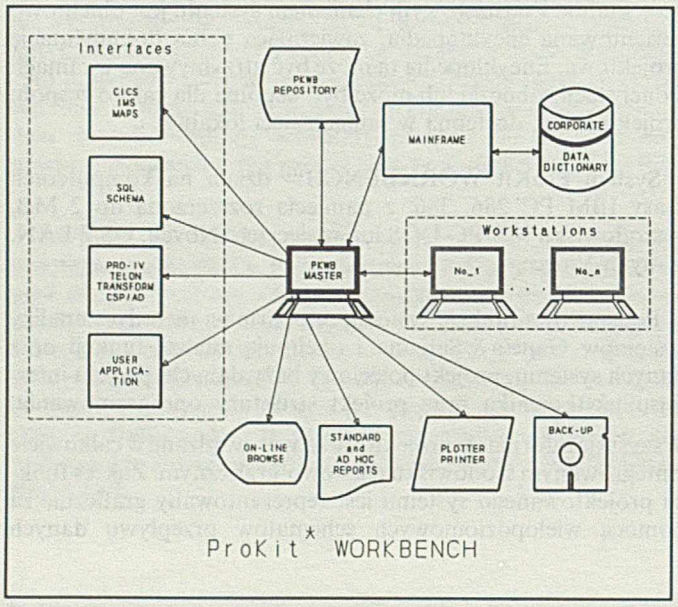
System ProKit*WORKBENCH® działa na komputerach klasy IBM PC 286, 386, z pamięcią rozszerzoną do 2 MB, w środowisku MS/PC-DOS lub w sieciach: Novell, OS/2 LAN, Banyan-Vinece.

Podstawowe funkcje systemu są oparte na metodyce analizy systemów Gane'a & Sarsona i obejmują analizę funkcji oraz danych systemu, projekt pojęciowy bazy danych, projekt interfejsu użytkownika oraz projekt struktury oprogramowania. Wszystkie informacje projektowe są wprowadzane w całkowicie zintegrowanym środowisku tekstowo-graficznym. Zakres funkcji projektowanego systemu jest reprezentowany graficznie za pomocą wielopoziomowych schematów przepływu danych



Rys. 5. Proces projektowania systemu użytkowego

(ang. *data flow diagrams*), a projekt pojęciowy bazy danych przy wykorzystaniu standardowej notacji Entity/Relationship (Chena) lub w postaci schematów Bachmana. Interfejs użytkownika jest prezentowany dynamicznie przez funkcje szybkiego prototypowania i symulacji jego działania, dając już na etapie projektu systemu możliwość weryfikacji funkcji konwersacyjnych. Projekt struktury oprogramowania jest reprezentowany graficznie za pomocą wielopoziomowych schematów strukturalnych (ang. *structure charts*).



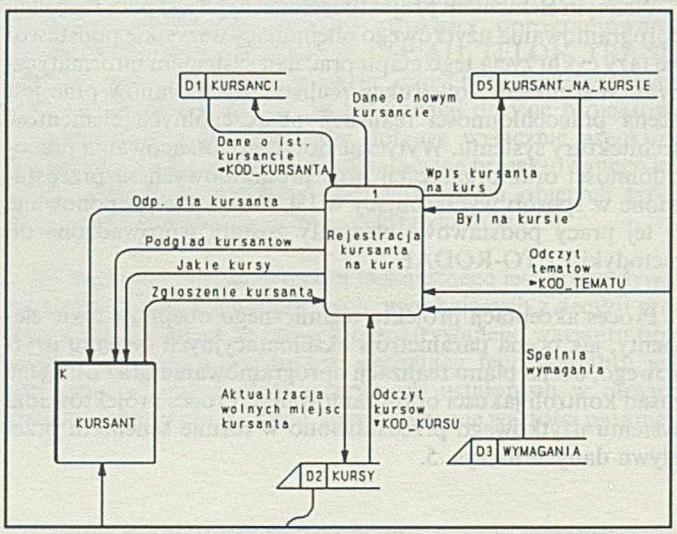
Rys. 6. Struktura systemu ProKit* Workbench

System ma również interfejsy do różnych środowisk systemowo–programowych. Projekt logiczny relacyjnej bazy danych jest – w oparciu o zasady odwzorowania pojęciowego modelu danych – generowany automatycznie w strukturę relacyjną, w standardzie ISO/SQL (dla Ingres, DB2, Oracle, Informix). Opisy struktur ekranów są również generowane automatycznie, zarówno jako deklaracje danych w językach programowania (Pascal, C, Cobol, Ada, Fortran, Basic, Assembler), jak i definicje map ekranów dla systemów CICS, IMS/DC.

Faza konstrukcji oprogramowania użytkowego jest wspomagana przez interfejsy do różnych generatorów aplikacji (PRO-IV®, CSP/AD, TELON, TRANSFORM), które przenoszą do tych środowisk większość informacji projektowych. System ma również bardzo bogate narzędzia do tworzenia raportów (standardowe, ad-hoc) wyprowadzanych na drukarkę lub do zbioru i dokumentacji graficznej rysowanej na ploterze.

Podstawowe cechy systemu:
- obiektowo zorientowana encyklopedia systemu,
- bogate środowisko graficzne, zintegrowane z encyklopedią systemu:
  - edytor schematów przepływu danych,
  - edytor pojęciowego modelu danych,
  - edytor schematów strukturalnych,
  - edytor diagramów prezentacyjnych,
- prototypowanie i symulacja interfejsu użytkownika,
- bogata dokumentacja projektowa:
  - raporty standardowe,
  - raporty ad-hoc,
  - dokumentacja graficzna,
- funkcje IMPORT-EXPORT,
- interfejsy do różnych generatorów aplikacji (PRO-IV®, CSP/AD, TELON, TRANSFORM),
- dostępna wersja sieciowa (ang. *Multi-user*),
- możliwa praca wielostanowiskowa (ang. *Multi-workstation*),
- obsługa dużych projektów (encyklopedia systemu do 128 MB),
- wielowersyjność projektu,
- pomoc w trybie on-line,
- obsługa różnych monitorów (mono, color), drukarek (igłowe, laserowe), ploterów (A0-A4).

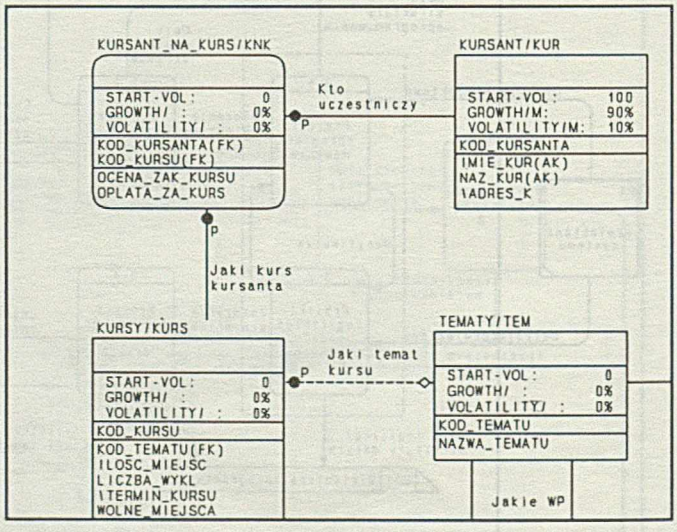
Podstawowymi modułami składowymi systemu ProKit* WORKBENCH® są: **ANALIZATOR, DATA MODELER, PROTOTYPER, DESIGNER.**



Rys. 7. Fragment schematu przepływu danych

Cechy modułu ANALIZATOR (rys. 7):

- 11-poziomowe schematy przepływu danych (Gane & Sarson),
- bilansowanie poziome i pionowe schematów,
- synchronizacja między modelem funkcji a modelem danych,
- maksimum 1600 symboli na jednym poziomie schematu.



Rys. 8. Fragment schematu pojęciowego modelu danych

Cechy modułu DATA MODELER (rys. 8):

- schematy obiekt-powiązanie (rozszerzony Chen lub Bachman),
- schematy relacyjne (rozszerzone IDEF1X),
- wspomaganie normalizacji z migracją kluczy,

- generowanie schematu logicznego relacyjnej bazy danych,
- synchronizacja między modelem danych a modelem funkcji,
- maksimum 1600 obiektów na schemacie modelu danych,
- możliwy pełny opis obiektów na schemacie (klucze, atrybuty, charakterystyka).

Cechy modułu PROTOTYPER (rys. 9):

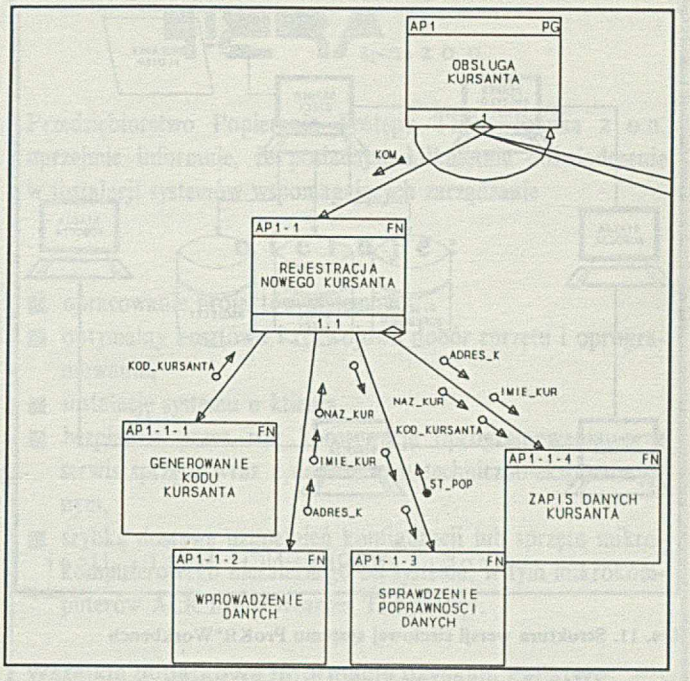
- projektowanie ekranów,
- symulacja interfejsu użytkownika poprzez tworzenie sekwencji ekranów, warunków poruszania się po nich i sposobu ich wypełniania,
- generowanie wersji źródłowych definicji danych w jednym z siedmiu języków (Pascal, C, Cobol, Ada, Fortran, Basic, Assembler),
- generowanie map ekranów dla CICS, IMS/DC,
- synchronizacja między ekranami prototypu a modelem funkcji,
- pełna dokumentacja graficzna i tekstowa zaprojektowanych ekranów.

Cechy modułu DESIGNER (rys. 10):

- wielopoziomowe schematy strukturalne (*Constantine*),
- wykorzystanie wyników analizy,
- pomoc przy specyfikacji programów dla różnych języków,
- maksimum 1600 obiektów na jednym poziomie schematu strukturalnego.

Cechy wersji sieciowej systemu (rys. 11):

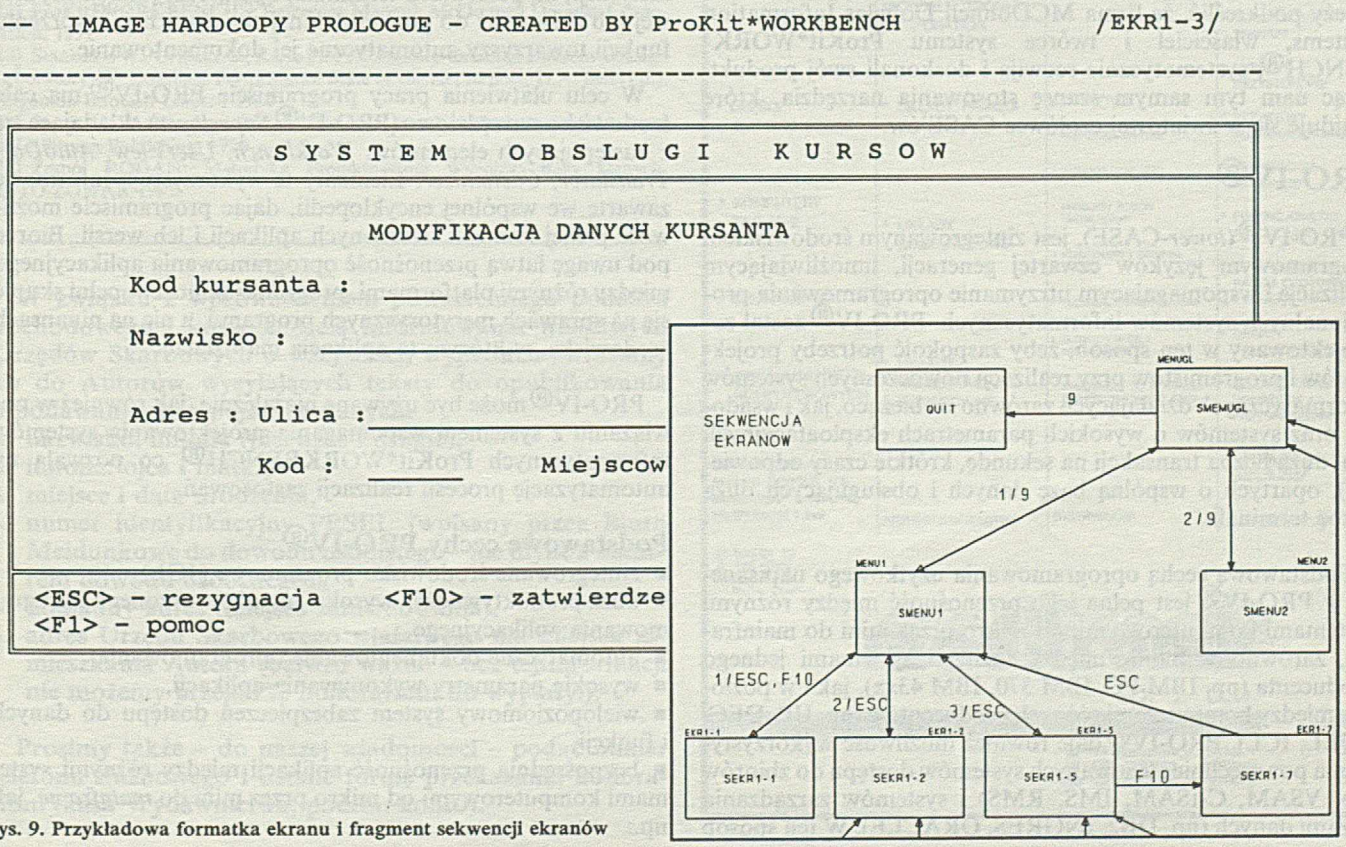
- architektura klient-serwer,
- równoległa obsługa wielu projektów i wersji w jednej sieci,
- synchronizacja pomiędzy encyklopedią a diagramami w czasie rzeczywistym,
- blokowanie na poziomie obiektów, a nie rekordów,
- log transakcji,
- mechanizm zapewniający atomowość transakcji modyfikujących encyklopedię projektu (ang. *page image posting*),



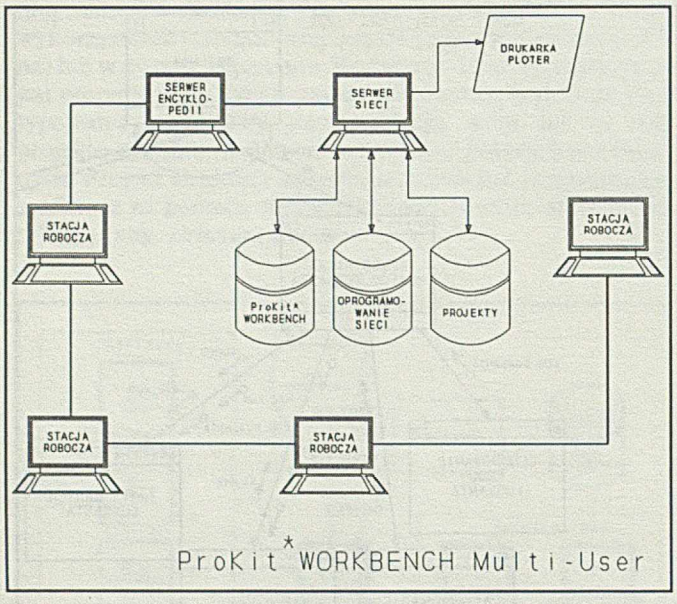
Rys. 10. Fragment schematu strukturalnego

- możliwość szybkiego odzyskiwania informacji projektowych (ang. *recovery*),
- równoważność funkcji użytkowych z wersją jednoużytkownikową,
- wszystkie cechy wersji dla jednego użytkownika.

Zastosowanie systemu ProKit*WORKBENCH® podnosi w sposób znaczący jakość realizowanych systemów informatycznych oraz znacznie obniża koszty opracowania. Wynika to z wprowadzenia sformalizowanej metodyki projektowania wspomaganej przez poszczególne funkcje systemu, usprawnie-



Rys. 9. Przykładowa formatka ekranu i fragment sekwencji ekranów



Rys. 11. Struktura wersji sieciowej systemu ProKit*Workbench

nia komunikacji między projektantami a użytkownikiem, a także w obrębie zespołu projektowego, oraz z możliwości łatwej prezentacji projektu użytkownikowi końcowemu. Kompletność dokumentacji projektowej oraz automatyczna kontrola spójności logicznej poszczególnych elementów projektu pozwala na wykrywanie błędów już we wczesnych fazach realizacji systemu. Dzięki takiemu narzędziu usprawnieniu ulega proces dokumentowania prac projektowych, a tym samym utrzymanie i konserwacja systemu stają się prostsze.

System ProKit*WORKBENCH® znalazł już zastosowania w wielu różnych jednostkach gospodarczych, takich jak: przedsiębiorstwa, instytuty naukowo-badawcze, uczelnie, banki, biura projektów zarówno w kraju, jak i za granicą. Jednocześnie należy podkreślić, że firma MCDonnell Douglas Information Systems, właściciel i twórca systemu ProKit*WORKBENCH®, systematycznie rozwija i doskonali swój produkt, dając nam tym samym szansę stosowania narzędzia, które znajduje się w światowej czołówce CASE'ów.

PRO-IV®

PRO-IV® (*lower*-CASE), jest zintegrowanym środowiskiem programowym języków czwartej generacji, umożliwiającym realizację i wspomagającym utrzymanie oprogramowania profesjonalnych systemów informatycznych. PRO-IV® został zaprojektowany w ten sposób, żeby zaspokoić potrzeby projektantów i programistów przy realizacji nowoczesnych systemów informatycznych działających zarówno na bieżąco, jak i wsadowo oraz systemów o wysokich parametrach eksploatacyjnych (np. duża liczba transakcji na sekundę, krótkie czasy odpowiedzi), opartych o wspólną bazę danych i obsługujących dużą liczbę terminali.

Podstawową cechą oprogramowania użytkowego napisanego w PRO-IV® jest pełna jego przenośność między różnymi systemami komputerowymi: od mikro przez mini do mainframe, zarówno w pionie między różnymi systemami jednego producenta (np. IBM PC, IBM 370, IBM 43xx), jak i w poziomie między komputerami różnych producentów (np. HP, DEC, BULL, ICL). PRO-IV® daje również możliwość wykorzystywania powszechnie stosowanych systemów dostępu do zbiorów (np. VSAM, C-ISAM, IMS, RMS) i systemów zarządzania bazami danych (np. DB2, INGRES, ORACLE). W ten sposób PRO-IV® zastępuje mniej wydajne i zależne od środowiska sprzętowo-programowego języki trzeciej generacji.

Wysoka efektywność programów użytkowych została uzyskana przez zastosowanie nowoczesnej architektury środowiska wykonawczego PRO-IV®, opartej na systemie modułów wielościeżkowych (ang. *re-entrant*), napisanych w Assemblerze lub C, stanowiących jądro PRO-IV® i realizujących funkcje elementarne (np. otworzenie zbioru, obsługa okienek, skrolowanie ekranu). Moduły te wraz z logiką systemu informatycznego, opisaną przy wyłorzystaniu komend języka PRO-IV®, tworzą wersję wykonawczą aplikacji. PRO-IV® nie tworzy ani skomplikowanego, ani interpretowanego kodu – wersja wynikowa jest sterowana tablicą. Takie podejście pozwoliło na ograniczenie interpretacji w czasie wykonywania do niezbędnego minimum, przy jednoczesnym zachowaniu wysokiego stopnia elastyczności oprogramowania użytkowego.

Podstawową jednostką programową i wykonawczą PRO-IV® jest funkcja (typu: MENU, SCREEN, RAPORT, UPDATE). Tworzenie funkcji użytkowych odbywa się przez wypełnianie uprzednio zdefiniowanych typów funkcji (ang. *fill-in-the-blank*) określając ich parametry. W ten sposób tworzymy moduły, które mogą być samodzielnie uruchamiane, testowane i modyfikowane. Tworzenie aplikacji polega na logicznym wiązaniu wcześniej stworzonych i przetestowanych modułów. Taki sposób podejścia znacznie ułatwia i przyspiesza tworzenie aplikacji, jak również umożliwia wielokrotne wykorzystywanie tych samych modułów. Przy opisywaniu modułów (funkcji użytkowych) określamy formatki ekranów lub wydruków, pola i zmienne na których pracuje moduł, zbiory danych, na których są wykonywane operacje odczytu i zapisu, logikę przetwarzania i komunikaty o błędach.

Dodatkowo istnieje możliwość definiowania wielopoziomowej struktury systemu zabezpieczeń (ang. *security*) polegającej na ustalaniu praw dostępu do danych i funkcji użytkowych. Prawa te mogą być określane dla poszczególnych działów instytucji, na poziomach terminala, operatora, funkcji użytkowej lub zbioru, aż do poziomu danej elementarnej. Tworzeniu funkcji towarzyszy automatyczne jej dokumentowanie.

W celu ułatwienia pracy programiście PRO-IV® ma całe środowisko narzędziowe (PRO-IV® *Superlayer*) składające się z następujących elementów: *Workbench, UserView, AutoDoc, Translator, Customizer*. Elementy te wykorzystują informacje zawarte we wspólnej encyklopedii, dając programiście możliwość pełnej kontroli tworzonych aplikacji i ich wersji. Biorąc pod uwagę łatwą przenośność oprogramowania aplikacyjnego między różnymi platformami, programista może w pełni skupić się na sprawach merytorycznych programu, a nie na niuansach środowiska, w którym ta aplikacja ma działać.

PRO-IV® może być używane niezależnie, jak również w powiązaniu z systemem wspomagania projektowania systemów informatycznych ProKit*WORKBENCH®, co pozwala na automatyzację procesu realizacji zastosowań.

Podstawowe cechy PRO-IV®

- zintegrowane środowisko programowe (4GL),
- duża produktywność i wysoka jakość stworzonego oprogramowania aplikacyjnego,
- automatyczne dokumentowanie oprogramowania,
- wysokie parametry wykonywania aplikacji,
- wielopoziomowy system zabezpieczeń dostępu do danych i funkcji,
- bezpośrednia przenośność aplikacji między różnymi systemami komputerowymi od mikro przez mini do *mainframe*, jak np.:

IBM SAA: MVS/VSE/VM, 43xx, 30xx, 937x, ES9000, AS/400
IBM AIX: 370, RS6000, PS/2
DIGITAL: VMS, ULTRIX
UNIX: AT&T, BULL, HEWLETT PACKARD, ICL, NCR, SUN, DATA GENERAL, NORSK DATA, PRIME, SIEMENS-NIXDORF, MCDONNEL DOUGLAS, FUJITSU
SPECIAL: BULL, FUJITSU, NOVELL, MS-DOS, SEQUENT, TOSHIBA

- interfejsy do różnych środowisk operacyjnych: C-ISAM, PRO-ISAM, VSAM, IMS, RMS, VM/CICS, CICS, VM/CMS, DL1, DB2, INGRES, ORACLE, SQL/DS, TSO, REALITY
- interfejs SQL do INGRES i ORACLE,
- pełna integracja z systemem wspomagania projektowania systemów informatycznych CASE ProKit*WORKBENCH®,
- zaawansowany język poleceń PRO-IV®,
- jądro PRO-IV® napisane w C lub Asemblerze zależnie od środowiska, umożliwiające efektywne wykorzystanie zasobów,
- szybkie prototypowanie funkcji użytkowych tworzących aplikacje,
- dostępna pełna wersja i wersja wykonawcza (ang. *run-time*).

LITERATURA

[1] Albano A., De Antonellis V., Di Leva A. (Eds): Computer-Aided Database Design – the DATAID Project. North-Holland 1985
[2] Ceri S. (Ed.): Methodology and Tools for Data Base Design. North-Holland 1983
[3] Dokumentacja techniczna Ingresu. Ingres Corporation, 1991
[4] Gane C., Sarson T.: Structured Systems Analysis: Tools & Techniques. McDonnell Douglas Corporation 1977
[5] Ghezzi C., Jazayeri M., Mandrioli D.: Fundamentals of Software Engineering. Prentice-Hall International Editions 1991
[6] Hammond J.L.: Performance Analysis of Local Computer Networks. Addison Wesley 1988
[7] Lazowska E.D., Zahorian J., Graham G.S., Sevcik K.C.: Quantitative System Performance – Computer System Analysis Using Queueing Network Models. Prentice-Hall 1984
[8] Mostardi T., Staniszkis W.: Multidatabase System Design Methodology. Database Technology, Vol. 3, No. 1
[9] Oleński J., Staniszkis W.: Projektowanie bazy danych. PWE 1984
[10] Olle T.W., Hagelstein J., Macdonald I.G., Rolland C., Sol H.G., Van Assche F.J.M., Verrijn-Stuart A.A.: Information Systems Methodologies – A Framework for Understanding. Addison Wesley 1991
[11] Orlando S., Perri V., Scrivano S., Staniszkis W.: Database Analyzer and Predictor – An Overview. Proc. Fifth Int. Conf. on Data Engineering, Los Angeles, February 1989, IEEE 1989
[12] ProKit*WORKBENCH® Reference Manual. McDonnell Douglas Corporation, 1991
[13] Staniszkis W., Orlando S., Rullo P.: Performance-Oriented Database Design Laboratory. W: Database Performance State of the Art Report, D.A. Bell (Ed.). Pergamon Infotech Limited 1984
[14] Staniszkis W.: Database Design w Practical Database Techniques, S.M. Deen (Ed.) Pitman Publishing, 1990
[15] Zespół RODAN: Metodyka Projektowania Wspólnej Bazy Danych. ZETO-ZOWAR 1985.

W związku z wprowadzeniem powszechnego podatku dochodowego i koniecznością informowania właściwych Urzędów Skarbowych o wszystkich wypłatach zwracamy się do Autorów wysyłających teksty do opublikowania o podawanie następujących danych:

- **nazwisko, imiona (pierwsze i drugie),**
- **imiona: ojca i matki,**
- **miejsce i data urodzenia,**
- **numer identyfikacyjny PESEL (wpisany przez Biuro Meldunkowe do dowodu osobistego – nie mylić z numerem dowodu osobistego!),**
- **dokładny adres (miejsce zameldowania),**
- **adres Urzędu Skarbowego właściwego dla miejsca zamieszkania Autora (bardzo ważne – bez tej informacji nie możemy przekazać honorarium do wypłaty!).**

Prosimy także – do naszej wiadomości – podać numer telefonu służbowego i ustalić formę przekazania honorarium (kasa Wydawnictwa, poczta, konto).

TERA Spółka Z O. O.

Przedsiębiorstwo Popierania Postępu TERA Spółka z o.o. uprzejmie informuje, że posiadając kilkuletnie doświadczenie w instalacji systemów wspomagających zarządzanie

o f e r u j e :

- opracowanie projektów systemów,
- optymalny kosztowo i rozwojowo dobór sprzętu i oprogramowania,
- instalację systemu u klienta,
- bezpłatnie przez rok konserwację oprogramowania oraz serwis sprzętu wraz z doradztwem techniczno-eksploatacyjnym,
- szybką dostawę uzupełnień konfiguracji lub sprzętu mikrokomputerowego niezależnego od systemu, w tym mikrokomputerów ALR firmy Wearnes Technology.

Wszelkie dodatkowe informacje uzyskają Państwo codziennie oprócz niedziel w Biurze Handlowym

40-025 Katowice, ul. Wojewódzka 31
tel. (faks): 155-26-72, teleks: 315448 tera pl

PAMIĘTAJ! Instalacje XENIX/NOVELL/PC MOS 386 oraz serwis to nasza specjalność – ZAPRASZAMY, ponieważ czterech lat doświadczeń nigdzie nie kupisz.

0/19/90

meditronik SPÓŁKA z o.o.
- CZĘŚCI ELEKTRONICZNE
- KOMPUTERY PS/1,PS/2
- DRUKARKI HP
- INSTALACJE SIECI KOMPUTEROWYCH

Partnerzy handlowi: ANALOG DEVICES, ITT, MOTOROLA, SAMSUNG, TELEFUNKEN i inni

UMC UNITED MICROELECTRONICS CORPORATION
- UKŁADY PAMIĘCI
- UKŁADY KOMPUTEROWE
- UKŁADY KOMUNIKACYJNE I KOMERCYJNE

PRZEDSTAWICIELSTWO

HEWLETT PACKARD COMPONENTS
- TRANSOPTORY
- WSKAŹNIKI ŚWIETLNE
- WYŚWIETLACZE LED
- PRODUKTY KODÓW KRESKOWYCH
- KONTROLERY I CZUJNIKI RUCHU
- TECHNIKA ŚWIATŁOWODOWA
- ELEMENTY W.CZ. I MIKROFALOWE
- PODZESPOŁY DO MONTAŻU POWIERZCHNIOWEGO (SMD)

DYSTRYBUCJA

BOURNS
- POTENCJOMETRY TRIMPOT
- HYBRYDY REZYSTOROWE
- REZYSTORY SUBMINIATUROWE
- BEZPIECZNIKI MULTIFUSE
- POTENCJOMETRY PRECYZYJNE
- POTENCJOMETRY PANELI CZOŁOWYCH I KODERY
- CEWKI I TRANSFORMATORY
- CZUJNIKI CIŚNIENIA, POŁOŻENIA I PRZYŚPIESZENIA

PMI

DYSTRYBUCJA

IBM Business Partner

meditronik sp. z o.o.

00-194 Warszawa,ul. Dzika 4
tel. (02) 6352263, 6352264
fax (02) 6352195, tlx 816075

SO/575/91

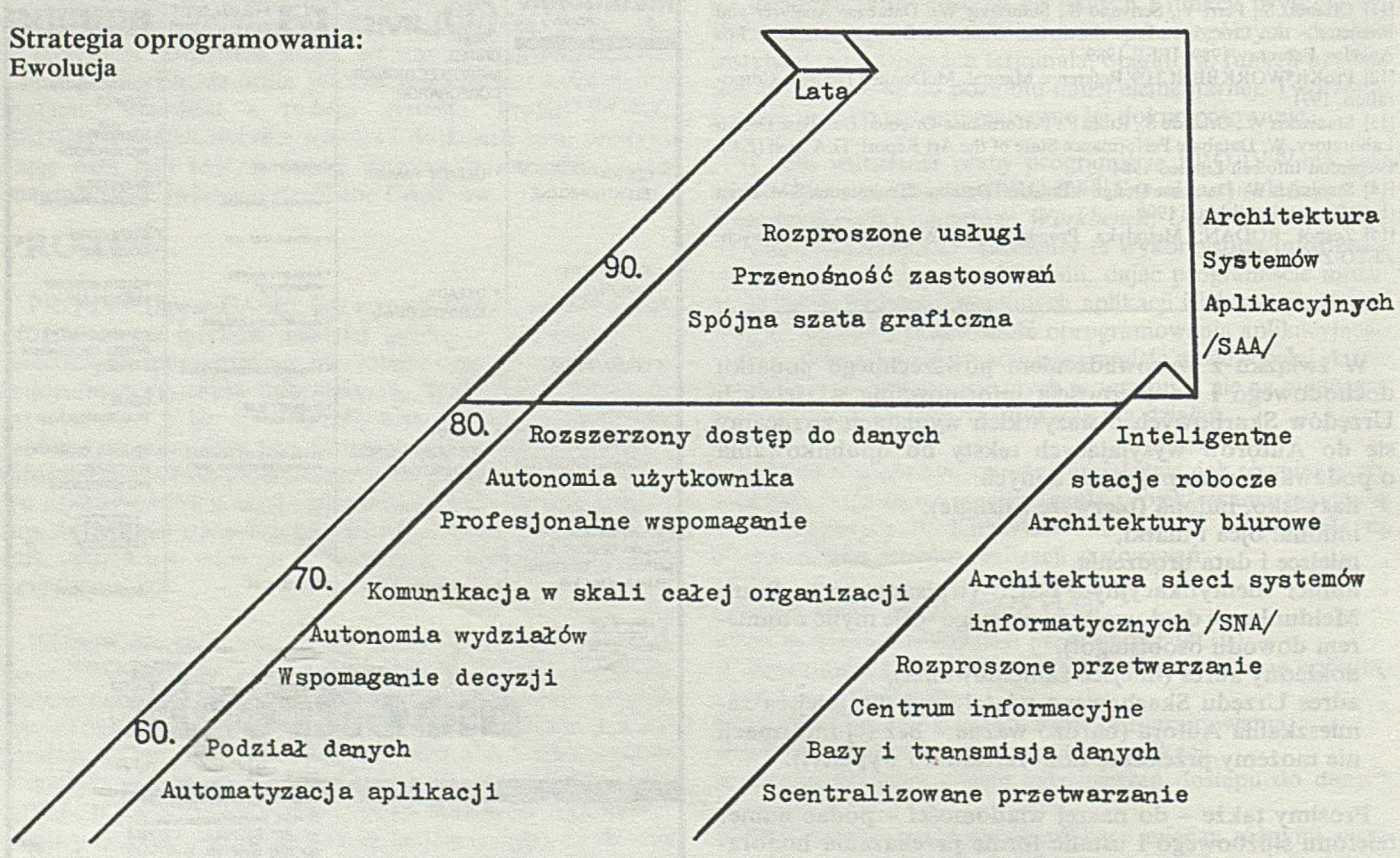
KRZYSZTOF MADRJAS
IBM Polska
Warszawa

AD/Cycle IBM-owska filozofia inżynierii projektowania oprogramowania

W artykule tym pragnę przedstawić założenia filozofii IBM w zakresie utrzymania spójności oprogramowania użytkowego dla głównych platform sprzętowych naszej firmy, oferowanych przez architekturę zwaną Architekturą Systemów Aplikacyjnych (*System Aplication Architecture* – SAA). Szybko rosnące wymagania dotyczące zwiększenia ilości przetwarzanych informacji wymogły konieczność stworzenia rodziny spójnych systemów operacyjnych, pozwalających na wykorzystanie jednej aplikacji w różnych środowiskach sprzętowych. Jednocześnie dąży się do rozmieszczenia funkcji i danych na terenie całego przedsiębiorstwa celem zaspokojenia rosnących oczekiwań użytkowników przy minimalnych nakładach na zmiany oprogramowania.

Na podstawie wymagań stawianych użytkowym systemom informacyjnym w przedsiębiorstwie zostaną pokazane możliwości oferowane przez architekturę SAA. Sformułowanie to definiuje bazę, na tle której działa IBM-owska strategia rozwoju oprogramowania użytkowego dla SAA o nazwie AD/Cycle (*Application Development* – Cykl Rozwoju Aplikacji).

AD/Cycle definiuje funkcje i środowisko przez cały cykl rozwoju oprogramowania – od momentu pierwszych rozmów z użytkownikiem do wdrażania i utrzymania działających aplikacji. Następnie to dzięki zestawowi narzędzi programowych produkowanych zarówno przez IBM jak i użytkowników oraz firmy współpracujące w tym zakresie z IBM. Narzędzia te zintegrowano na płaszczyźnie projektowania aplikacji.



Rys. 1. Zmiany architektury i technologii systemów komputerowych od lat sześćdziesiątych do dnia dzisiejszego

Po przedstawieniu założeń SAA oraz AD/Cycle zostaną przedstawione wybrane produkty programowe, reprezentujące tę właśnie strategię tworzenia oprogramowania użytkowego.

Ewolucja systemów informatycznych

Przedstawiony niżej historyczny przegląd technologii przetwarzania informacji od lat sześćdziesiątych do dnia dzisiejszego pokazuje, jak zmieniały się w tym okresie wymagania użytkowników oraz w jaki sposób firma IBM odpowiadała na nie zmianami architektury i technologii systemów komputerowych.

W latach sześćdziesiątych dominowało przetwarzanie zcentralizowane, nastawione na obsługę głównych systemów użytkowych przedsiębiorstwa związanych z rachunkiem ekonomicznym i materiałowym. Wraz ze wzrostem liczby automatyzowanych funkcji rosło zapotrzebowanie na udoskonalenie zasobów pamięci, w której przechowywano informacje, a także metod dostępu do tych zasobów. Odpowiedzią IBM było zaoferowanie baz danych oraz metod komunikacyjnych, umożliwiających wprowadzenie „transparentności danych". Umożliwiło to rozwój zasobów pamięci dyskowych użytkowników przy minimalnych nakładach na modyfikację oprogramowania.

W latach siedemdziesiątych użytkownicy zaczęli wykorzystywać informacje w sposób bardziej efektywny, redukując nieaktywny czas pracy aplikacji. Sytuacja ta wymogła konieczność stworzenia oprogramowania wspomagającego podejmowanie decyzji. Odpowiadając na to zapotrzebowanie IBM opracował koncepcję Centrum informacyjnego. Udostępniło ono użytkownikom spójny zestaw narzędzi programowych, wspomagających podejmowanie decyzji przez projektantów oprogramowania użytkowego oraz użytkowników.

Równocześnie u wielu użytkowników powstało zapotrzebowanie na autonomiczne, lokalne przetwarzanie danych, nie wymagające ingerencji centralnego ośrodka przetwarzania w przedsiębiorstwie. Stanowiło to podstawę do wprowadzenia technologii przetwarzania rozproszonego.

Rosnące i coraz bardziej wyrafinowane potrzeby użytkowników wymagały efektywnego sterowania dostępem do informacji. Konsekwencją tego rozwoju było opracowanie przez IBM architektury sieci systemów informatycznych (*System Network Architecture* – SNA) oraz wprowadzenie „transparentności komunikacji". Dziś SNA to architektura komunikacyjna, wykorzystywana przez użytkowników do zarządzania komunikacją informacji przez różnorodne lokalizacje systemów przetwarzania danych.

W latach osiemdziesiątych IBM kontynuował rozwój rozwiązań systemowych poprzednich dekad. Została opracowana koncepcja systemów przetwarzania biurowego, umożliwiających tworzenie i dystrybucję dokumentów w całym przedsiębiorstwie. Celem zaspokojenia rosnących potrzeb indywidualnych użytkowników wprowadzono model inteligentnej (programowalnej) stacji roboczej (*Programmable WorkStation* – PWS) w postaci komputera osobistego (PC). Stały rozwój oprogramowania umożliwił użytkownikom powszechny dostęp do profesjonalnych narzędzi programowych.

Obecnie bardziej niż kiedykolwiek główny nacisk położono na obsługę pojedynczego użytkownika na tle całego przedsiębiorstwa. Pracownicy chcą mieć bezpośredni dostęp do informacji oraz aplikacji, i to niezależnie od ich lokalizacji. Chcą oni także wykorzystywać te same funkcje i w taki sam sposób, lecz na różnych systemach komputerowych. Równocześnie chcą rozdzielać funkcje na różne systemy celem optymalnego wykorzystania specyficznych możliwości różnego sprzętu i oprogramowania systemowego.

Aby wykorzystać ograniczone zasoby w sposób bardziej efektywny, konieczne jest projektowanie takich aplikacji, które można wykorzystywać w różnorodnych środowiskach systemowych i sprzętowych i które mogą przetrwać zmiany w założeniach organizacyjnych przy minimalnych zmianach programowych.

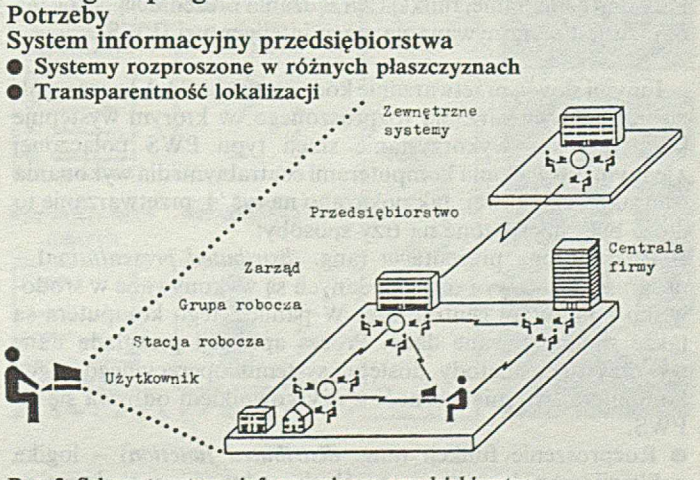
Rozwiązania przetwarzania informacji na lata dziewięćdziesiąte muszą rozwiązywać totalne potrzeby wszystkich grup użytkowników przedsiębiorstwa – personalne, grupowe, działowe, lokalizacyjne – dostawców i klientów.

Celem jest więc opracowanie rozwiązania obejmującego wszystkie te wymagania, działającego w różnorodnych środowiskach sprzętowych oraz odpowiadającego jednocześnie na różnorodne indywidualne zapotrzebowanie użytkowników. Rozwiązanie takie powinno zapewnić przejrzysty dostęp do danych i aplikacji w przedsiębiorstwie – niezależnie od ich fizycznej lokalizacji.

Dla przykładu, użytkownicy pracujący na jednym poziomie indywidualnie, mający dostęp do danych lokalnych, wykorzystują je i przetwarzają, a następnie rozprowadzają przez całe przedsiębiorstwo, wypełniając zawarte w nich polecenia. Takie zastosowanie wymaga zarówno unikatowych zadań, jak i wspólnych funkcji, takich jak np. zarządzanie kalendarzem i tworzenie raportów.

Użytkownicy biorą także udział w pracach grupowych, wymagających do ich wykonania unikalnych zadań, lub zadań dzielonych jednocześnie z innymi grupami. Grupa tworzy wspólnie organizację zlokalizowaną w jednym lub wielu miejscach. Oprócz tego powinno być możliwe dzielenie informacji, choć nie wszystkich, o dostawcach i klientach. Dla obsługi wszystkich tych powiązań jest konieczny bardzo sprawny system komunikacyjny. Sieć przetwarzania danych musi zapewnić użytkownikom dostęp do tych danych i funkcji, których oni wymagają oraz dostarczyć rezultat ich pracy tym, którzy tego oczekują.

Sieć taka musi obsłużyć wiele rodzajów informacji: tekst, grafikę, głos i obraz, integrując różnorodne urządzenia. IBM oferuje sprzęt i oprogramowanie dla całego spektrum wymienionych wymagań, proponując koncepcję systemu informacyjnego przedsiębiorstwa (EIS – *Enterprise Information System*) – rys. 2.

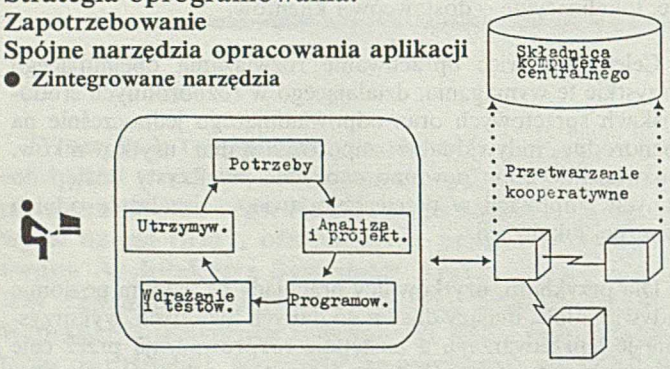


Rys. 2. Schemat systemu informacyjnego przedsiębiorstwa

Środowisko tworzenia aplikacji

Dla zapewnienia projektantom możliwości tworzenia aplikacji dla systemu informacyjnego przedsiębiorstwa jest konieczne dostarczenie im rodziny zintegrowanych narzędzi programowych do obsługi pełnego cyklu czasu trwania aplikacji. Oznacza to, że narzędzia te muszą działać w rozproszonym środowisku i tworzyć wspólne zasoby projektowe, a osoby projektujące, kodujące, testujące i obsługujące oprogramowanie użytkowe wymagają jako stanowiska pracy inteligentnych stacji roboczych (rys. 3).

Strategia oprogramowania:
Zapotrzebowanie
Spójne narzędzia opracowania aplikacji
- **Zintegrowane narzędzia**

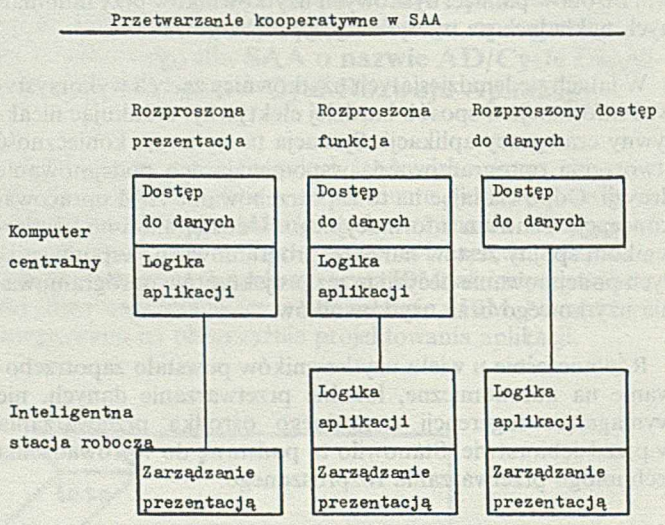


Rys. 3. Współczesna technologia opracowania aplikacji

Część takich funkcji, jak np. dzielona składnica (ang. *repository*), powinno rezydować w zasobach komputerów centralnych, podczas gdy inne, takie jak np. grafika czy projektowanie, można doskonale realizować na stacjach roboczych. Lokalizacja funkcji powinna być przejrzysta dla użytkownika.

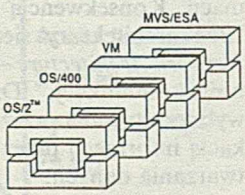
Użytkownik systemu informacyjnego powinien mieć dostęp do zasobów położonych w dowolnym miejscu przedsiębiorstwa objętego siecią informacyjną lub w sieci zewnętrznej. Tego typu pracę zapewnia technologia opracowywania aplikacji przedstawiona na rys. 3. Przetwarzanie kooperatywne SAA (ang. *cooperative processing*) zapewnia użytkownikowi zintegrowany i prosty dostęp do wszystkich zasobów komputerowych. Przez zasoby te rozumiemy: dane, sprzęt komputerowy (np. drukarki o dużej wydajności przyłączone do systemów centralnych), pakiety programów użytkowych lub programy specjalne, niedostępne na pojedynczych stacjach. Użytkownik powinien mieć możliwość wykorzystywania tych zasobów bez potrzeby posiadania wiedzy o ich typie i lokalizacji. W tym samym czasie zasoby te muszą być chronione przez specjalne programy zabezpieczające. Przetwarzanie kooperatywne ma jedną postać, chociaż wykonuje się w dwu lub wielu połączonych systemach. Przy tym co najmniej funkcja zarządzania prezentacją graficzną obrazu jest wykonywana na stacji inteligentnej (PWS).

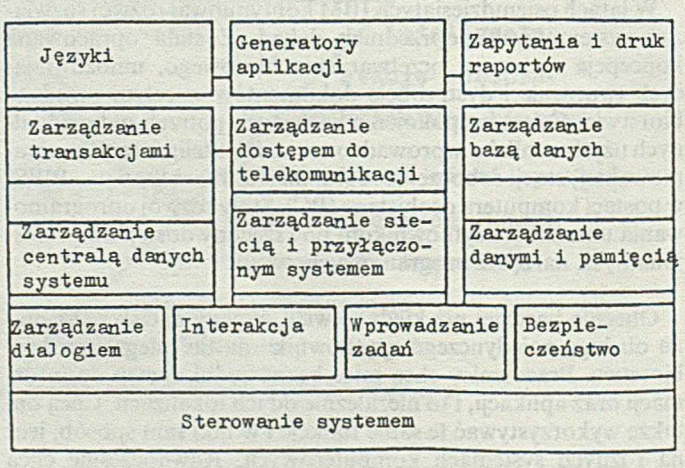
Innymi słowy, przetwarzanie kooperatywne SAA to szczególna postać przetwarzania rozproszonego, w którym występuje koordynowane wykorzystanie stacji typu PWS połączonej z jednym lub wieloma komputerami centralnymi dla wykonania transakcji użytkowej. Jak pokazano na rys. 4, przetwarzanie to może być rozproszone na trzy sposoby:
- Rozproszona prezentacja (ang. *distributed presentation*) – program użytkowy i obsługa danych są wykonywane w środowisku komputera centralnego. W pamięci tego komputera są także zapamiętywane dane. Proces aplikacji przegląda dane wykorzystując metody dostępu systemu operacyjnego tegoż komputera. Jedynie łączność z użytkownikiem odbywa się na PWS.
- Rozproszenie funkcji (ang. *distributed function*) – logika aplikacji jest wdrożona w sposób skoordynowany zarówno na PWS, jak i w komputerze centralnym. Wykorzystywane dane mogą być umieszczone także w obu środowiskach. Na życzenie użytkownika proces aplikacji odszukuje dane dla ich odzysku i zapisu w tej samej lokalizacji. Funkcja prezentacji jest wykonywana na PWS.
- Dostęp do rozproszonych danych (ang. *distributed data access*) – cała aplikacja od strony logicznej jest umiejcowiona na PWS. Środowisko komputera centralnego może być wykorzystywane jako serwery danych. Jeżeli systemy zarządzania bazą danych (ang. *Data Base Management System* – DBMS) w obu środowiskach działają w sposób kooperatywny, aplikacja na PWS może wykorzystywać dane z zewnętrznych komputerów centralnych przez skierowanie zapytania do lokalnego DBMS. Lokalizacja oraz inne szczegóły o danych spoza systemu lokalnego nie muszą być specyfikowane w programie użytkowym na PWS. Ten sposób dostępu do danych został wdrożony w architekturze rozproszonych relacyjnych baz danych (ang. *Distributed Relational Database Architecture* – DRDA). W tym przypadku przez środowisko komputerów centralnych o funkcjach serwerów rozumie się nie tylko komputery System/370 i System/390, ale także AS/400 i PS/2.



Rys. 4. Schemat przetwarzania kooperatywnego w SAA

Strategia oprogramowania:
Spójna struktura oprogramowania
- **Rozszerzona na wszystkie systemy struktura systemu operacyjnego**





Rys. 5. Struktura systemów operacyjnych SAA

Ogłoszony w marcu 1987 r. wstęp do SAA zdefiniował interakcyjny model aplikacyjny dla następujących systemów operacyjnych: OS/2, OS/400, VM/CMS, MVS/ESA. Komponenty, z których składają się te systemy można przedstawić w trzech grupach (rys. 5). Są to:

- płaszczyzna tworzenia aplikacji i zarządzania komunikacją,
- zarządzanie bazami danych (identyczne dla wszystkich platform sprzętowych w architekturze SAA) i metodami dostępu do danych,
- płaszczyzna styku ze sprzętem – specyficzna dla każdej platformy sprzętowej, gdyż od niej zależy wykorzystanie charakterystycznych cech poszczególnych typów komputerów.

AD/CYCLE – podstawowe definicje

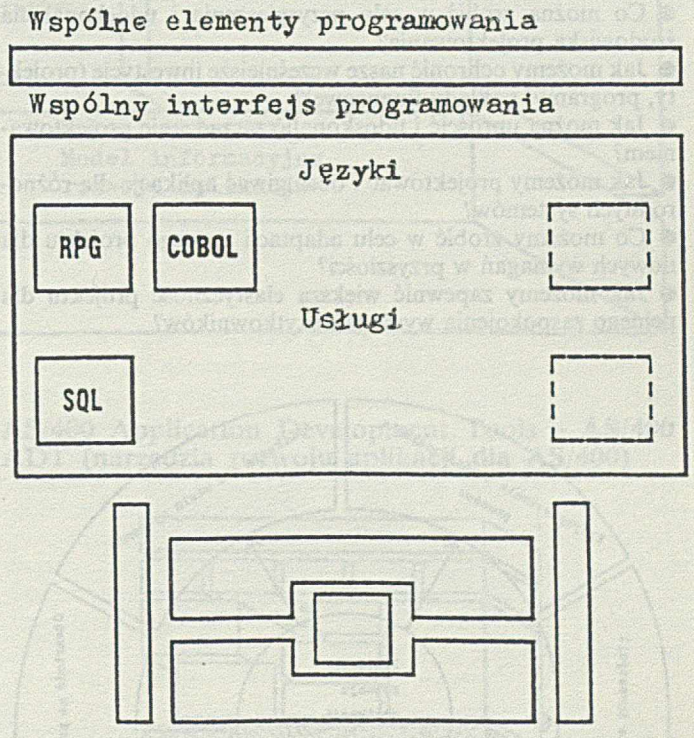
Aby umożliwić tworzenie jednorodnych aplikacji, architektura SAA wnosi standaryzację podstawowych połączeń w tej rodzinie systemów. Najbardziej znaczące komponenty to: wspólny dostęp do użytkownika (ang. *Common User Access* – CUA), jednolite usługi komunikacyjne (ang. *Common Communication Support* – CCS) oraz standard połączeń międzyprogramowych (ang. *Common Program Interface* – CPI).

CUA definiuje podstawowe elementy komunikacji z użytkownikiem – reguły wspólnej terminologii, wspólnego wyglądu ekranów, identycznych menu i innych interaktywnych technik, umożliwiających jednolity wygląd i wspólną atmosferę dla różnych aplikacji i systemów (co przedstawiono na rys. 6). Podstawowy cel CUA, to ułatwienie obsługi programów i procesów uczenia się.

CPI obejmuje języki programowania, komendy i odwołania wykonywane przez programy. Dzieli się na dwie główne kategorie: języki i usługi programowe. Standard ten (przedstawiony na rys. 7) definiuje założenia, umożliwiające wykorzystanie aplikacji napisanych w jednym środowisku systemowym w innych systemach operacyjnych.

Podstawowe języki programowania AD/CYCLE to: COBOL, RPG, C, FORTRAN, PL/1, REXX oraz CSP. Zostały zdefiniowane także elementy obsługi baz danych – strukturalny język zapytań (ang. *Structure Query Language* – SQL) i środowisko zarządzania zapytaniami (ang. *Query Management Facility* – QMF). Umożliwia to stworzenie jednolitego środowiska obsługi baz danych na wszystkich płaszczyznach sprzętowych. CPI definiuje także elementy dialogu między użytkownikami a aplikacjami oraz sposób komunikacji między programami. Przez wprowadzenie definicji komunikacji dla języków wysokiego poziomu (ang. *High Level Language* CPI – HLL CPI) umożliwia on prostą wymianę danych i funkcji między aplikacjami napisanymi w różnych językach.

CCS jest wykorzystywany do łączenia aplikacji, systemów, sieci i urządzeń zgodnie z rys. 8. Komponent ten definiuje reguły połączeń zgodne z międzynarodowymi standardami na różnych płaszczyznach. Podstawowy poziom komunikacyjny to sieci i linie komunikacyjne (protokoły transmisji) wykorzystywane przez wszystkie systemy IBM. Poziom średni, to sterowanie sesjami, komunikacja między programami itp. Na najwyższym poziomie znajduje się obsługa strumieni danych oraz usługi aplikacyjne, umożliwiające wykorzystanie usług komunikacyjnych opartych na transmisji zdefiniowanej przez niższe poziomy – w programach aplikacyjnych.

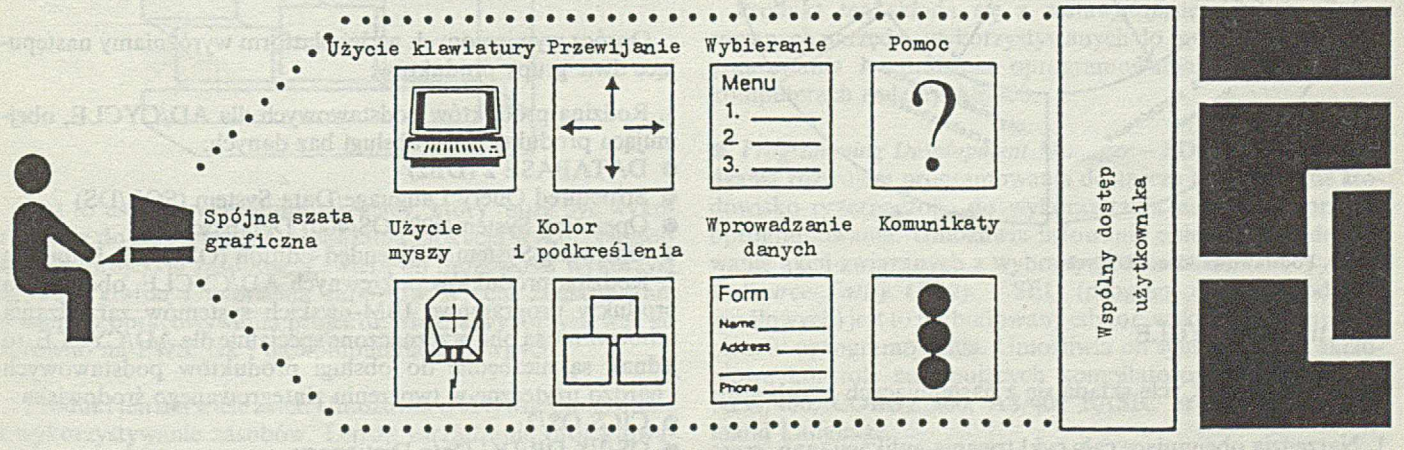


Rys. 7. Współpraca programów w SAA

Architektura Systemów Aplikacyjnych: Wspólny dostęp użytkownika

Perspektywa projektanta:

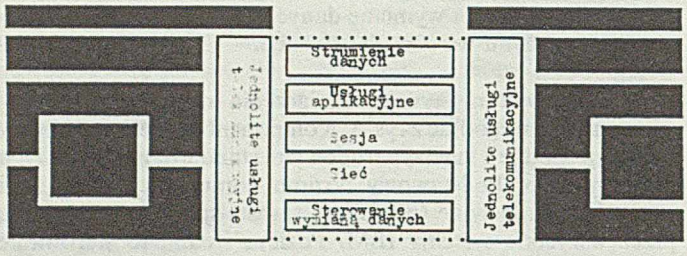
- **Spójna szata graficzna we wszystkich aplikacjach**
- **Stacje robocze, zarówno inteligentne, jak i nieinteligentne**



Rys. 6. Podstawowe elementy standardu CUA

Architektura Systemów Aplikacyjnych: Jednolite usługi telekomunikacyjne

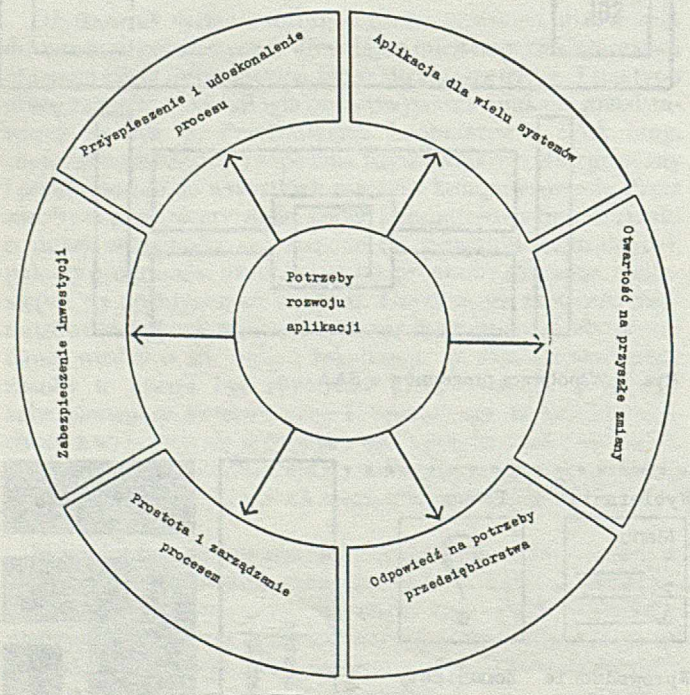
- Ustanowienie SNA oraz międzynarodowych standardów
- Komunikacja między systemami i aplikacjami



Rys. 8. Zasady komunikacji w SAA

Idee AD/CYCLE (rys. 9) można zdefiniować przez następujące pytania, dobrze znane dzisiejszym projektantom oprogramowania użytecznego:

- Co można zrobić w celu przyspieszenia i udoskonalenia środowiska projektowania?
- Jak możemy ochronić nasze wcześniejsze inwestycje (projekty, programy, nakłady finansowe)?
- Jak można uprościć i udoskonalić zarządzanie projektowaniem?
- Jak możemy projektować i obsługiwać aplikacje dla różnorodnych systemów?
- Co możemy zrobić w celu adaptacji naszego projektu dla nowych wymagań w przyszłości?
- Jak możemy zapewnić większą elastyczność projektu dla pełnego zaspokojenia wymagań użytkowników?



Rys. 9. Podstawowa idea AD/Cycle

Struktura AD/CYCLE

Struktura AD/Cycle składa się z następujących trzech grup komponentów:

1. Narzędzia obejmujące cały cykl trwania aplikacji (ang. *cross life cycle tools*), które dostarczają zintegrowanych funkcji, umożliwiających, dzięki specjalnym produktom IBM, zarządzanie i sterowanie całym procesem projektowania aplikacji, a mianowicie:
   - zarządzanie projektami i procesami,
   - analizy impaktowe,
   - przygotowanie dokumentacji,
   - rekonstruowanie procesów i projektów.

2. Narzędzia tworzenia aplikacji, dostarczające podstawowych funkcji koniecznych przy projektowaniu aplikacji:
   - modelowanie danych, procesów, kierunków zmian założeń,
   - analiz i projektowania dla baz danych, logiki, ekranów, raportów,
   - budowy prototypów w celu stworzenia początkowego obrazu aplikacji, obejmujące:
     - języki i narzędzia do edycji, kompilacji i analizy błędów,
     - generatory, które przenoszą projektowanie z poziomu języków trzeciej generacji na poziomy najwyższe,
     - systemy baz danych wiedzy dla projektowania i wspomagania narzędzi projektowych,
     - możliwości testowania i modelowania,
     - wspomaganie obsługi pracy i możliwość rekonstrukcji oprogramowania użytkowego.

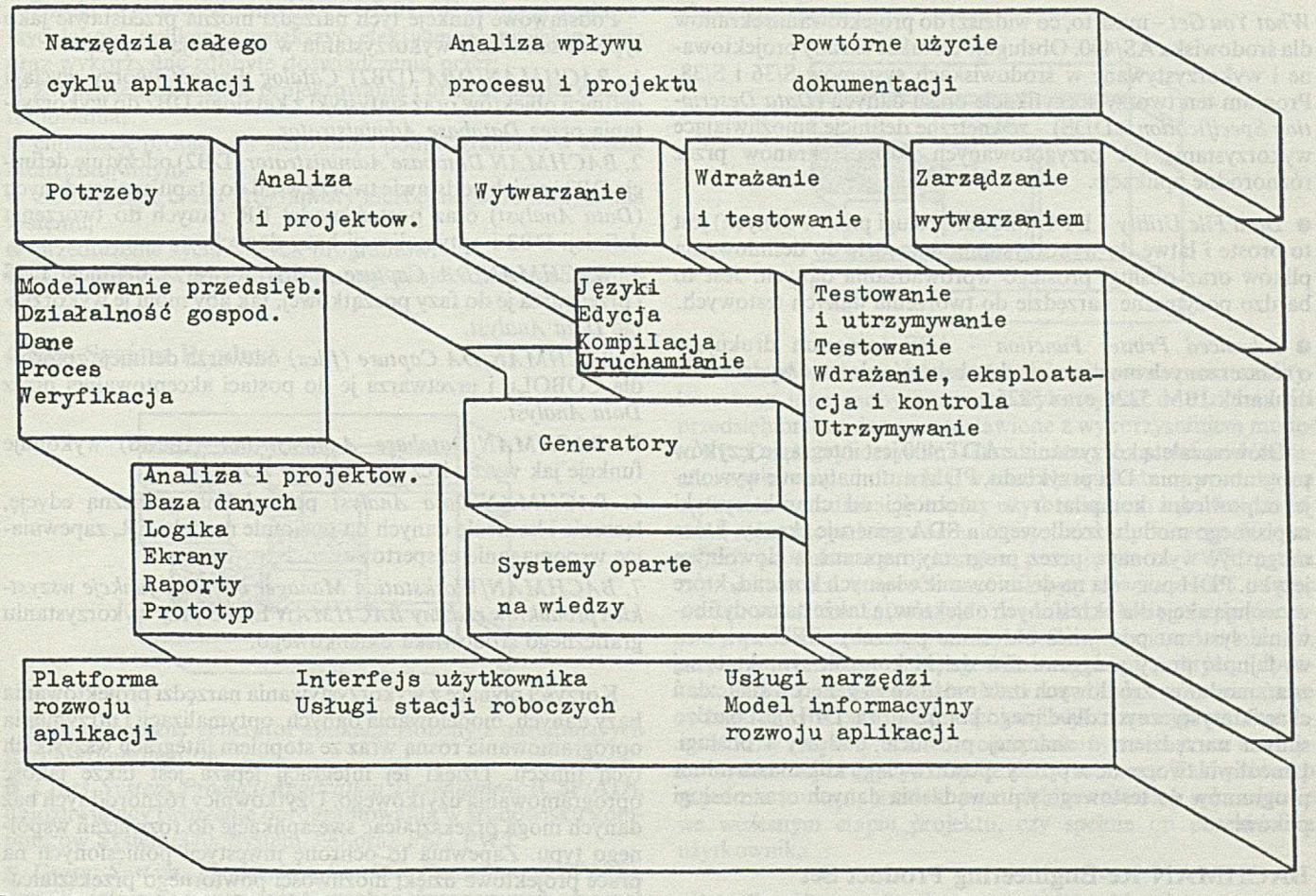
3. Platforma projektowania aplikacji integruje wszystkie działania związane ze środowiskiem projektowania przez:
   - CUA – jednolity wygląd i wrażenia estetyczne dla projektantów, niezależność poszczególnych funkcji narzędziowych; stacja robocza AD/Cycle jest zgodna z graficznym modelem SAA CUA,
   - usługi oferowane przez stację roboczą (ang. *workstation services*); opierając się na standardzie CUA umożliwiają użytkownikowi wykorzystanie znacznych możliwości graficznych komputera PS/2,
   - usługi oferowane przez narzędzia (ang. *tools services*); jest to zestaw funkcji, których wymagają narzędzia programowe w celu zapewnienia integracji danych i innych wspólnych funkcji,
   - usługi oferowane przez składnicę (ang. *repository services*); obsługa składnicy oferuje usługi umożliwiające aktualizację danych, utrzymanie jednolitych poziomów poprawek itp.,
   - informacyjny model projektowania aplikacji (ang. *application development information model*); spójny obraz definicji projektowania systemu użytkowego, takich jak: dane, programy, ekrany – łącznie z elementami definicji dostępnych funkcji,
   - dwuczęściowy model informacyjny (ang. *two part information model*), obejmujący:
     - model ER (ang. *entity-relationship*), oznaczający wystąpienie-relację, dla definiowania relacji niskiego poziomu,
     - model obiektowy (ang. *object model*).

Elementy składowe cyklu tworzenia oprogramowania zostały przedstawione na rys. 10.

Oprócz wymienionych wyżej platform wyróżniamy następujące dwie grupy produktów:

1. Rodzina produktów podstawowych dla AD/CYCLE, obejmująca produkty SAA obsługi baz danych:
   - DATABASE 2 (DB2)
   - Structured Query Language/Data System (SQL/DS)
   - Operating System/400 (OS/400) Database
   - Operating System/2 Extended Edition (OS/2 EE) Database

2. Rodzina produktów pokrewnych AD/CYCLE, obejmująca produkty programowe IBM-owskich systemów zarządzania i chociaż nie są one przeznaczone specjalnie dla AD/CYCLE, to jednak są niezbędne do obsługi produktów podstawowych i bardzo użyteczne w tworzeniu zintegrowanego środowiska:
   - CICS OS/2
   - OS/VS DB/DC Date Dictionary
   - Database Relational Application Directory (DBRAD)

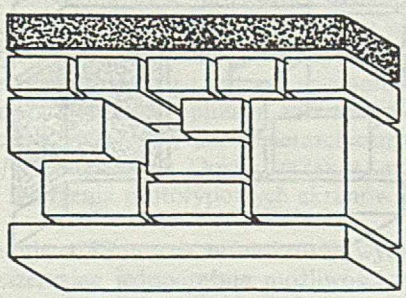


Rys. 10. Elementy składowe koncepcji tworzenia oprogramowania

- Operating System/2 Extendid Edition (OS/2 EE)
- OS/2 Programming Tools and Information Version 1, 2
- Query Management Facility (QMF)

Przegląd wybranych produktów AD/CYCLE

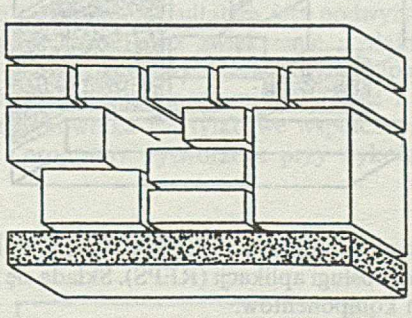
Aplication System (AS) Project Management



Jest to usługowy program sterujący, który może być wykorzystany do efektywnego zarządzania alokacją wykorzystywanych w projekcie zasobów. Zawiera on możliwości wykorzystywania grafiki i tworzenia raportów w celu zilustrowania postępów opracowywania projektu. Można wykorzystywać go zarówno na PWS, jak i na komputerach głównych.

Produkt ten ma wiele zalet. Umożliwia efektywne planowanie i wykorzystywanie zasobów. Lepsze zarządzanie projektowaniem aplikacji pozwala zwiększyć wydajność projektowania, a tym samym zmniejszyć jego koszty.

AS/400 Application Development Tools – AS/400 ADT (narzędzia rozwoju aplikacji dla AS/400)



Produkt ten składa się z zestawu następujących zintegrowanych narzędzi, wykorzystywanych do projektowania, programowania i wdrażania oprogramowania użytkowego na komputerach rodziny AS/400.

- *Programming Development Manager* – PDM (system zarządzania rozwojem programowania dostarcza zintegrowane środowisko przeznaczone do wykorzystania narzędzi tworzenia oprogramowania. Umożliwia listowanie obiektów i podejmowanie akcji związanych z wybranymi obiektami.
- *Source Entity Utility* – SEU (program obsługi modułów źródłowych) jest to rozbudowany edytor, wykorzystywany przy pisaniu oprogramowania. Umożliwia on syntaktyczną i składniową kontrolę następujących kompilatorów: AS/400 PL/1, RPG/400, COBOL/400, AS/400 BASIC oraz AS/400 Command Language.
- *Screen Design Aid* – SDA (moduł projektowania ekranów) to program wykorzystujący metodę WYSIWYG (*What You See Is*

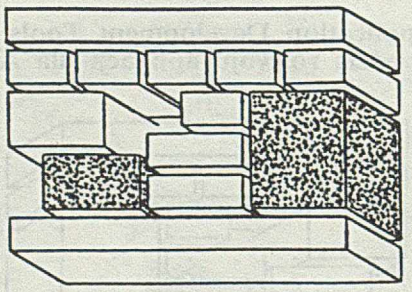
What You Get – masz to, co widzisz) do projektowania ekranów dla środowiska AS/400. Obsługuje on także ekrany projektowane i wykorzystywane w środowiskach systemów S/36 i S/38. Program ten tworzy specyfikacje opisu danych (*Data Description Specification* – DDS) – zewnętrzne definicje umożliwiające wykorzystanie tak przygotowanych postaci ekranów przez różnorodne aplikacje.

- *Data File Utility* – DFU (moduł obsługi plików danych) jest to proste i łatwe do wykorzystania narzędzie do definiowania plików oraz obsługi prostego wprowadzania danych. Jest to bardzo pożyteczne narzędzie do tworzenia danych testowych.

- *Advanced Printer Function* – APF (program drukujący o rozszerzonych możliwościach) obsługuje złożone wydruki dla drukarek IBM 5224 oraz 5225.

Główną zaletą korzystania z ADT/400 jest integracja języków programowania. Dla przykładu, PDM automatycznie wywołuje odpowiedni kompilator w zależności od charakterystyki napisanego modułu źródłowego a SDA generuje ekrany, które mogą być wykonane przez programy napisane w dowolnym języku. PDH pozwala na definiowanie własnych komend, które wywołują akcję dla określonych obiektów, a także na modyfikowanie systemu pod ściśle określone potrzeby. SEU zwiększa wydajność pracy programistów dzięki kontroli syntaktycznej zdań modułów źródłowych oraz możliwości generowania zdań charakterystycznych dla danego kompilatora. DFU jest bardzo silnym narzędziem o znacznej prostocie budowy i obsługi. Umożliwia tworzenie w prosty sposób w ciągu kilkunastu minut programów do testowego wprowadzania danych oraz obsługi plików.

BACHMAN Re-Engineering Product Set



Jest to zestaw produktów firmy BACHMAN przeznaczony do tworzenia i obsługi aplikacji (REPS). Składa się z następujących siedmiu komponentów:

- *BACHMAN/DATABASE ADMINISTRATOR* (DB2),
- *BACHMAN/Data Analyst,*
- *BACHMAN/DA Capture* (IMS),
- *BACHMAN/DA Capture (files),*
- *BACHMAN/Database Administrator* (IDMS),
- *BACHMAN/DBA* (DB2) *Catalog Extract,*
- *BACHMAN/Workstation Manager.*

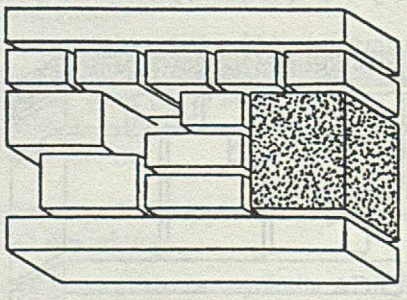
Narzędzia te obejmują kilka faz cyklu rozwoju aplikacji. Mogą być one wykorzystywane zarówno w początkowej fazie rozwoju aplikacji na etapie zbierania zapotrzebowań w przedsiębiorstwie oraz na etapie analizy i projektowania, jak i w fazie projektowania oprogramowania oraz baz danych. Umożliwiają one także nadzorowanie i modyfikowanie już eksploatowanych systemów. Charakteryzują się unikalną cechą – możliwością zastosowania produkcyjnego systemu użytkowego korzystającego z IMS, IDMS lub DB2 lub plików obsługiwanych przez standardowe metody dostępu jako wejście informacyjne do projektów nowych systemów użytkowych opartych na DB2.

Podstawowe funkcje tych narzędzi można przedstawić jako typową sekwencję wykorzystania w następujący sposób:

1. *BACHMAN/DBA* (DB2) *Catalog Extract* tworzy wyciągi definicji obiektów oraz statystyki z katalogu DB2 do wykorzystania przez *Database Administrator.*
2. *BACHMAN Database Administrator* (DB2) odczytuje definicje DB2 i na ich podstawie tworzy wsad do etapu analizy danych (*Data Analyst*) oraz postać model ER danych do tworzenia definicji DB2 i optymalizacji bazy danych.
3. *BACHMAN/DA Capture* (IMS) odtwarza definicje IMS i przetwarza je do fazy początkowej, tak aby mógł je wykorzystać *Data Analyst.*
4. *BACHMAN/DA Capture (files)* odtwarza definicje zbiorów dla COBOLu i przetwarza je do postaci akceptowalnej przez *Data Analyst.*
5. *BACHMAN/Database Administrator* (IDMS) wykonuje funkcje jak wyżej, lecz dla zbiorów IDMS.
6. *BACHMAN/Data Analyst* pozwala na graficzną edycję, łączenie i kontrolę danych na poziomie modelu ER, zapewniając wspomaganie ekspertowe.
7. *BACHMAN/Workstation Manager integruje funkcje wszystkich produktów rodziny BACHMAN* ERPS przy wykorzystaniu graficznego środowiska okienkowego.

Korzyści płynące z wykorzystywania narzędzi projektowania bazy danych, modelowania danych, optymalizacji i utrzymania oprogramowania rosną wraz ze stopniem integracji wszystkich tych funkcji. Dzięki tej integracji lepsza jest także jakość oprogramowania użytkowego. Użytkownicy różnorodnych baz danych mogą przekształcać swe aplikacje do rozwiązań wspólnego typu. Zapewnia to ochronę inwestycji poniesionych na prace projektowe dzięki możliwości powtórnego przekształcania i modyfikowania (ang. *re-engineering*) aplikacji już eksploatowanych. Efektem zastosowania tego zestawu produktów jest także skrócenie czasu przygotowania systemu użytkowego w wyniku wzrostu wydajności pracy oraz zastosowania najnowszych technik informatycznych z dziedziny baz wiedzy. Funkcje dostarczane przez te narzędzia można wykorzystywać na PWS z rodziny OS/2.

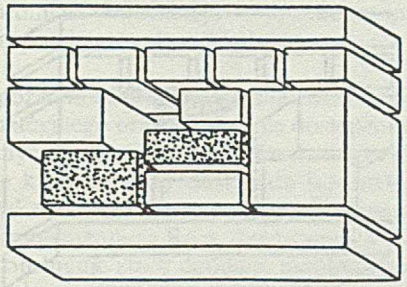
COBOL Structuring Facility



Moduł strukturalizacji COBOLu (*COBOL Structuring Facility*) to program umożliwiający modyfikację istniejących programów, napisanych przy wykorzystaniu starych kompilatorów, do postaci w pełni strukturalnego języka COBOL II. Jeżeli narzędzie to jest wykorzystywane łącznie z modułem konwersji poziomu komend (COBOL/CICS *Command Level Conversion Aid* – CCCA), wówczas program ten, wykorzystując techniki sztucznej inteligencji, modyfikuje stare oprogramowanie (łącznie z transakcjami konwersacyjnymi) do postaci nowoczesnych strukturalnych modułów. Miejsca niemożliwe do automatycznej restrukturyzacji są analizowane przy konwersacyjnej współpracy z projektantem lub programistą.

Dzięki zastosowaniu modułu COBOL SF możemy podwyższyć jakość aplikacji, zwiększyć efektywność projektowania oraz wykorzystać zdobyte doświadczenia przez:
- zmniejszenie kosztów projektowania i utrzymywania oprogramowania,
- eliminację problemów sterowania podprogramami w kodzie niestrukturalnym,
- zmniejszenie czasu programisty niezbędnego do utrzymania systemu,
- przedłużenie życia starych programów,
- łatwiejsze tworzenie dokumentacji projektowej i programowej.

Cross System Product

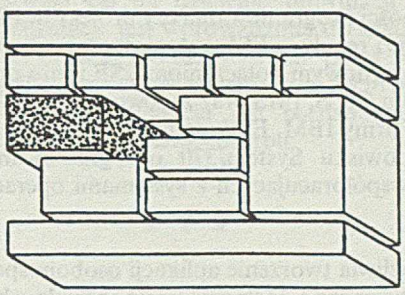


CSP to potężny generator aplikacji złożony z następujących trzech komponentów:
- *Cross System Product/Application Development* (CSP/AD), umożliwiający tworzenie oprogramowania w środowisku komputerów System/370, System/390 oraz PS/2 PWS.
- *Cross System Product/Application Execution* (CSP/AE), umożliwiający wykonywanie transakcji wygenerowanych przez CSP w różnych środowiskach systemowych (nie obejmuje on jednak środowiska IMS/VS.
- *Cross System Product/370 Runtime Services* (CSP/370 RS) generuje moduły źródłowe w COBOLu na podstawie modułów CSP oraz umożliwia wykonywanie skompilowanych modułów w środowisku IMS/VS.

CSP jest sztandarowym generatorem aplikacji dla architektury SAA. Produkt ten umożliwia pracę projektantów przy tworzeniu definicji systemu aplikacyjnego na pojedynczych stacjach roboczych lub w środowisku lokalnej sieci komputerowej (LAN). Zapewnia wiele możliwości strumieniowego definiowania, testowania i dokumentowania oraz wsadowego lub konwersacyjnego wykonywania nowych systemów użytkowych a także interakcyjny nadzór i testowe uruchomianie aplikacji. CSP wspomaga tworzenie aplikacji zarówno dla relacyjnych (DB2, SQL/DS, SQL/400), jak i hierarchicznych (IMS/DB, DL/1, DOS/VS) baz danych. Dostarcza także narzędzi przeznaczonych do tworzenia prototypowych ekranów i aplikacji.

Wykorzystując CSP można zwielokrotnić wydajność projektantów, rozszerzając jednocześnie możliwości wykorzystania projektowanej aplikacji na wszystkich dostarczanych w ramach SAA płaszczyznach sprzętowych-systemowych. Jako logiczny komponent AD/Cycle, CSP umożliwia redukcję kosztów oraz rozszerzenie możliwości tworzenia i modyfikowania systemów użytkowych przez uproszczenie procesu rozwoju aplikacji. Możliwość tworzenia prototypów aplikacji we wczesnych etapach projektowania pozwala znacznie podwyższyć jakość przygotowywanych produktów oraz lepiej usatysfakcjonować użytkownika. Dzięki umożliwieniu przetwarzania na głównych systemach komputerowych firmy IBM – CSP zapewnia zachowanie dotychczasowych inwestycji sprzętowych oraz dalszy rozwój oprogramowania.

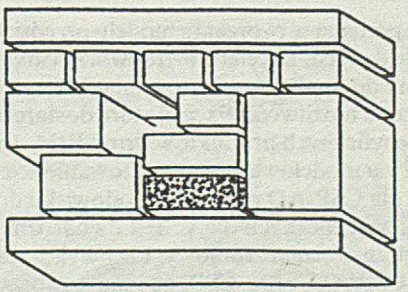
DevelopMate



DevelopMate zaprojektowano dla stosowania przez analityków oraz inne osoby pragnące wykorzystać do działalności przedsiębiorstwa dane przedstawione z wykorzystaniem modelu ER. Dzięki wykorzystaniu standardowych metod modelowania i zapisywaniu informacji w centralnej składnicy produkt ten upraszcza komunikację między użytkownikiem a profesjonalistami, zajmującymi się tworzeniem systemu użytkowego. Komunikacja produktu z projektantem pracującym na PWS została zaprojektowana w sposób zapewniający uproszczenie i pomoc we wprowadzaniu informacji o modelu działania przedsiębiorstwa w postaci tekstowej. Analityk definiuje dane o działalności, atrybuty i charakterystyki zgodnie z regułami, polityką i dynamiką rozwoju firmy. DevelopMate umożliwia tworzenie z tego rodzaju danych prototypów aplikacji wykorzystując standardowe ekrany i raporty, z możliwością dokonywania przez analityka dodatkowych modyfikacji na stacji roboczej. Taki prototyp zapewnia kontrolę modelu przedsiębiorstwa oraz sprawdzenie we wczesnym etapie projektu, czy spełnia on oczekiwania użytkownika.

Możliwość sprawdzenia poprawności aplikacji we wczesnym etapie pozwala zwykle znacznie zmniejszyć koszty oraz ograniczyć liczbę pojawiających się problemów. Dzięki włączeniu sprzężenia z użytkownikiem niemal od początku tworzenia aplikacji, można w sposób prawie doskonały zaspokoić jego oczekiwanie i żądanie. Rezultatem jest podwyższenie jakości powstającego systemu oraz zwiększenie wydajności zespołu tworzącego system użytkowy. DevelopMate zapewnia funkcje będące logicznymi komponentami AD/Cycle i ochraniające inwestycje użytkownika nie tylko we współczesny sprzęt, ale także dane i programy wytworzone przy wykorzystaniu tego sprzętu.

Expert System Environment



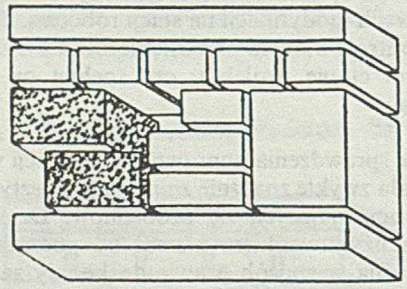
Środowisko systemu ekspertowego (*Expert System Environment* – ESE) można wykorzystywać w różnorodny sposób. Najczęściej program ten jest wykorzystywany do rozwiązywania problemów:
- diagnostycznych,
- klasyfikacyjnych,
- pomocy na żądanie (ang. *help desk*),
- doradztwa finansowego.

Użytkownicy mogą wykorzystywać ten program bezpośrednio do tworzenia systemów użytkowych, ponieważ nie wymaga on umiejętności programowania czy projektowania. Wystarczy podać relacje i reguły, a ESE automatycznie utworzy aplikacje. Dzięki standardowym połączeniom ESE może pracować także w środowisku CICS, IMS oraz zapewnia dostęp do relacyjnych baz danych firmy IBM. ESE ma także możliwości projektowania w środowisku System/370 oprogramowania dla stacji roboczych współpracujących z systemami operacyjnymi DOS lub OS/2.

ESE umożliwia tworzenie aplikacji osobom spoza środowiska informatycznego i dostarcza wiele specjalnych opcji, takich jak:
- proste w korzystaniu funkcje projektowania ekranów,
- możliwość graficznego obrazowania danych i tworzenia histogramów,
- proste w korzystaniu metody dostępu do danych zapamiętanych w relacyjnych bazach danych (DB2 i SQL/DS).

ESE pomaga wykorzystać wysoko produktywne narzędzia tworzenia aplikacji łącznie z możliwością tworzenia prototypów. Dzięki uproszczeniu nadzoru nad działającymi aplikacjami, ESE umożliwia zmniejszenie kosztów utrzymania oprogramowania użytkowego.

Index Technology Tools



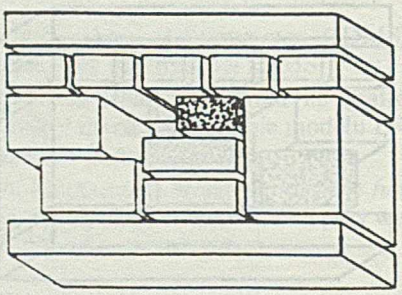
Narzędzia firmy Index Technology (*Index Technology Tools*) służy do automatyzacji dwóch pierwszych faz cyklu modyfikacji oprogramowania użytkowego. Składają się one z następujących modułów:
- PC Prism,
- Excelerator/IS,
- Customizer,
- Excelerator dla DB2,
- Excelerator dla CSP/AD.

PC Prism pomaga w tworzeniu modelu przedsiębiorstwa oraz w przepływie danych. Excelerator/IS tworzy całkowity warsztat przeznaczony do analiz i projektowania. Firma Index Technology rozszerzyła możliwości Exceleratora dostarczając wspomaganie dla specyficznych produktów firmy IBM. Excelerator dla DB2 umożliwia modelowanie i projektowanie baz danych DB2. Excelerator dla CSP/AD dostarcza środowisko do projektowania aplikacji specyficznych dla CSP z dwudrożnym łącznikiem przez zewnętrzny format źródłowy. Przy wykorzystaniu Exceleratora dla CSP/AD oraz CSP, można utworzyć kompletną aplikację, którą następnie można wykonywać w środowisku CSP. Excelerator dostarcza także narzędzi pomocniczych do tworzenia prototypów. Również ekrany mogą być projektowane w czasie projektowania.

Customizer pozwala dostosować parametry środowiska projektowania do specyficznych wymogów konkretnego przedsiębiorstwa. Moduł ten może być wykorzystany do modyfikowania słowników oraz do tworzenia nowych relacji, unikalnych menu, grafiki, ikon itp.

Narzędzia firmy Index Technology, które mają charakter narzędzi typu CASE, oferują różnorodne korzyści. Jeżeli projektant może w prosty sposób modyfikować środowisko swej pracy, to prowadzi do lepszego wykorzystania czasu pracy oraz stwarza możliwości osiągnięcia optymalnych efektów projektowania. Użytkownik może wpływać nie tylko na modyfikację środowiska tworzenia aplikacji, ale także wykorzystywać najbardziej efektywne technologie, wybierając optymalne rozwiązanie dla uzyskania produktu końcowego najlepszej jakości. Dzięki wykorzystaniu modułu Excelerator dla DB2 oraz Excelerator dla CSP/AD projektant tworzący aplikację ma możliwość integracji kilku faz cyklu modyfikowania aplikacji.

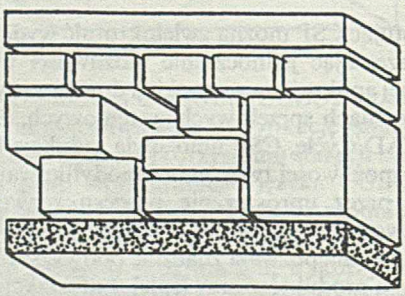
Inspect



Inspekt wspomaga interaktywne lub wsadowe testowanie oraz analizowanie błędów dla programów napisanych w językach C/370 oraz PL/1, łącznie z aplikacjami napisanymi w obu językach jednocześnie. Testowane programy mogą być śledzone z dokładnością do poszczególnych zdań źródłowych modułu. Możliwe jest modyfikowanie oraz definiowanie nowych zmiennych, a także zmiany w logice programu. Przy wykorzystaniu wielokrotnych okien wszystkie informacje są podawane w bardzo prosty i klarowny sposób. Możliwe jest asynchroniczne modyfikowanie prędkości, pozwalające na indywidualne strojenie środowiska testowego.

Testowanie i usuwanie błędów z nowoczesnych aplikacji jest często żmudnym zajęciem, pochłaniającym dużo czasu, zwłaszcza gdy programy użytkowe są napisane w więcej niż jednym języku. *Inspect* to nowoczesny, bardzo mocny produkt programowy firmy IBM. Oszczędza czas programistów konieczny do uruchamiania i testowania nowych programów dzięki szybkiemu i efektywnemu rozwiązywaniu problemów wynikających z błędów zawartych w programach użytkowych. Produkt ten dostarcza technologii umożliwiającej zwiększenie wydajności projektowania i programowania. Jest to prosty w korzystaniu logiczny komponent AD/CYCLE.

Repository Manager/MVS (RM/MVS)



Składnica (ang. *repository*) może stać się wspólnym miejscem składowania wszystkich informacji dotyczących projektowania aplikacji w przedsiębiorstwie. Każda informacja jest zapamięty-

wana tylko jeden raz i tylko w jednym miejscu, a dostęp do niej jest dzielony przez wszystkie narzędzia stosowane do projektowania i modyfikowania oprogramowania użytkowego przez cały cykl jego trwania. Wykorzystanie modelu ER (*entity-relationship* = wystąpienie-relacja) oraz technik służących do modelowania opisu danych o informacji w składnicy, a także specyfikacji charakterystyk logiki aplikacji, stwarza możliwości zaawansowanych metod modelowania.

Repository Manager zapewnia także obsługę łączników obiektowych, gdzie dane i związane z nimi akcje są zapamiętywane jako typy obiektów i związane z nimi metody. Narzędzia składnicy mogą pobierać informacje ze składnicy odwołując się do danych za pomocą usług modelu ER lub usług obiektowych. Dane obiektowe można zapamiętywać wewnątrz składnicy lub poza nią, natomiast informacje – łącznie z informacjami kontrolnymi zapamiętanymi w składnicy.

RM/MVS pozwala na kontrolowanie zmian zapamiętanego modelu informacyjnego oraz sterowanie dostępem do informacji zdefiniowanych przez ten model. Przez usługi obiektowe jest możliwy także kontrolowany dostęp do informacji rezydujących w bibliotekach oraz w innych zbiorach. *Repository Manager*/MVS jest implementacją SAA *Repository* CPI w środowisku ESA. CPI obsługuje zbiór definicji, manipulowanie i sterowanie informacjami zapamiętanymi w składnicy.

RM/MVS umożliwia utworzenie centralnego punktu sterowania i kontroli dla wszystkich informacji o przedsiębiorstwie oraz projektowanej aplikacji. Większość tych danych jest przechowywana w składnicy. Dane zapamiętane w odseparowanych (zewnętrznych) zbiorach, takie jak np. moduły ładowalne, mogą także pozostawać pod kontrolą RM/MVS. Produkt ten wspomaga tworzenie modelu danych przedsiębiorstwa, zawierający wszystkie niezbędne do budowy aplikacji dane. Model ten pozwala zwiększyć wydajność tworzenia aplikacji przez:

- redukcję redundancji danych,
- redukcję problemów definicji danych,
- tworzenie modeli prototypowych.

RM/MVS jest jednym z głównych komponentów logicznych AD/Cycle. Pozwala on na dzielenie informacji zawartych w składnicy. Informacja jest wprowadzona do składnicy tylko jeden raz, a następnie wykorzystywana przez różnorodne programy narzędziowe. RM/MVS zapewnia także bezpieczeństwo danych, których integracja jest bardzo prosta dzięki jednemu centralnemu punktowi kontroli.

* * *

Podstawowym celem ogłoszenia SAA oraz AD/Cycle nie był zamiar dostarczenia gotowych narzędzi do tworzenia i modyfikowania oprogramowania użytkowego, lecz stworzenie szkieletu systemu tworzenia aplikacji przez zdefiniowanie standardów połączeń i podstawowych definicji.

Szkielet ten, to fundament budowy narzędzi do tworzenia aplikacji. Ogłaszając założenia IBM zaproponował współpracę wszystkim firmom tworzącym oprogramowanie. Charakterystycznym zjawiskiem dla tej dziedziny jest fakt, że wiele spośród omówionych w tym artykule produktów pochodzi nie z IBM, a z firm będących naszymi partnerami przy tworzeniu oprogramowania AD/Cycle. Firmy te są objęte wspólną nazwą *International Aliances*.

SA obejmuje tylko niektóre płaszczyzny sprzętowe oferowane przez firmę IBM. Dla systemów nie objętych SAA, tzn. VSE oraz AIX, obecnie nie jest oferowane tak kompleksowe rozwiązanie. Jednak już dziś wiele z produktów, oferowanych dla systemów objętych SAA, może pracować także pod kontrolą VSE lub AIX. Również wiele standardowych złączy programowych ma już dziś swoje odpowiedniki w tych systemach, umożliwiając prosty transfer danych i zbiorów. Dalsza integracja tych środowisk będzie postępowała wraz z upływem czasu.

Sądzę, że AD/Cycle wytycza drogę w przyszłość, Przyszłość, w której użytkownik nigdy nie będzie sam, ponieważ zawsze będzie mu towarzyszył IBM.

Konkurs im. Jerzego Trybulskiego

Polskie Towarzystwo Informatyczne ogłasza trzecią edycję Ogólnopolskiego Konkursu im. Jerzego Trybulskiego na najlepsze wdrożone prace z dziedziny zastosowań informatyki w gospodarce narodowej

Warunki Konkursu

★ Zgłoszone na konkurs prace mogą dotyczyć rozwiązań informatycznych z dziedziny zastosowań wdrożonych w jednostkach gospodarki narodowej (uspołecznionej i nieuspołecznionej) w kraju i za granicą.
★ W konkursie mogą wziąć udział obywatele Polski, którzy są autorami prac wykonanych na terenie kraju.
★ Będą oceniane tylko rozwiązania informatyczne bieżąco eksploatowane przez użytkownika co najmniej jeden rok.
★ Zgłoszone prace będą oceniane według kryterium uzyskanych wymiernych efektów ekonomicznych.
★ Nie będą oceniane wdrożone rozwiązania informatyczne dotyczące: języków programowania, systemów operacyjnych, szeroko rozumianej technologii komputerowego przetwarzania, oprogramowania narzędziowego i pomocniczego.

Wymagania dotyczące zgłoszenia

★ Prawo zgłaszania na konkurs prac spełniających warunki regulaminowe mają dyrektorzy, kierownicy jednostek autorskich lub eksploatacyjnych oraz indywidualni autorzy rozwiązań informatycznych.
★ Prace konkursowe należy zgłaszać na formularzu „Karta zgłoszenia", który wraz z instrukcją wypełniania można otrzymać w Oddziałach i Kołach PTI.
★ Do karty zgłoszenia należy dołączyć krótką dokumentację rozwiązania oraz opinię bezpośredniego użytkownika zawierającą: czas trwania eksploatacji (w tym datę rozpoczęcia bieżącej eksploatacji), ocenę funkcjonalną, ocenę łatwości eksploatacji (użytkowania), wady i zalety rozwiązania, efekty wymierne i niewymierne, jakie uzyskał użytkownik itp.
★ Przedłożona opinia musi być zatwierdzona przez dyrektora – kierownika jednostki gospodarczej użytkującej system.

Termin i adres nadsyłania zgłoszeń

Dokumentację prac konkursowych spełniających wyżej podane warunki i wymagania należy zgłaszać w terminie do 30 listopada 1992 roku pod adresem:

Polskie Towarzystwo Informatyczne, Oddział Dolnośląski, ul. Włodkowica 8, p. 18, 50-072 Wrocław, tel. 44-52-79, oraz wpłacić kwotę 3 mln złotych na konto Dolnośląskiego Oddziału PTI: PKO BP IV O/Wrocław, nr 93549-123350-132.

Szczegółowe informacje dotyczące warunków, nagród, organizacji konkursu itp. są podane w regulaminie opublikowanym w „Biuletynie PTI" nr 9, 1985; dysponują nim także Oddziały i terenowe Koła PTI.

Dolnośląski Oddział
Polskiego Towarzystwa Informatycznego

Kim był Jerzy Trybulski?

Na początku lat 60. istniała w środowisku wrocławskim zorganizowana grupa entuzjastów elektronicznej techniki obliczeniowej, której

przewodniczył Jerzy Trybulski. Z inicjatywy tej grupy w 1964 r. utworzono pierwszy w Polsce ogólnostępny, usługowy ośrodek elektronicznej techniki obliczeniowej we Wrocławiu. Pionierską misję zorganizowania ośrodka powierzono Jerzemu Trybulskiemu, którego losy od tej chwili były ściśle związane z polską informatyką. Przedsiębiorstwo to, prowadzone ofiarnym wysiłkiem garstki ludzi, zadziwiło swym rozwojem największych nawet optymistów.

Czternastoletni okres działalności Jerzego Trybulskiego jako dyrektora ZETO we Wrocławiu, a następnie ponad dwuletni okres kierowania Stołecznym Ośrodkiem Informacji i Techniki Obliczeniowej – to niemal nieustanne pasmo osiąganych wielkim wysiłkiem sukcesów w dziedzinie informatyki i pozyskiwania rynku, w realizacji śmiałych przedsięwzięć obejmujących coraz to nowe obszary zastosowań implikujących rozwój bazy sprzętowej i technologii przetwarzania informacji. Jerzy Trybulski przykładał również ogromną wagę do spraw wychowania i kształcenia kadr informatycznych.

W Komitecie Założycielskim Polskiego Towarzystwa Informatycznego Jerzy Trybulski pełnił funkcję przewodniczącego Komisji Rejestracyjnej. Nie dane Mu było wzięcie udziału w Zjeździe Założycielskim PTI w dniu 23 maja 1981 r. Niespodziewana choroba i jej tragiczny finał 23 kwietnia 1981 r., na miesiąc przed zarejestrowaniem naszego Towarzystwa, zaskoczyły i pogrążyły w żalu środowisko informatyczne.

Osiągnięcia zawodowe zapewniają Jerzemu Trybulskiemu trwałe miejsce w historii polskiej informatyki. Dla upamiętnienia tego faktu – na mocy uchwały I Walnego Zjazdu PTI – Ogólnopolski Konkurs na najlepsze wdrożone prace z dziedziny zastosowań informatyki w gospodarce narodowej nazwano Jego imieniem.

OŚRODEK POSTĘPU TECHNICZNEGO w Katowicach
zaprasza w dniach 15–18 września 1992 r.
na tradycyjne już

VI MIĘDZYNARODOWE TARGI OPROGRAMOWANIA

SOFTARG'92

Podstawową ofertą SOFTRAG-u'92 jest oprogramowanie komputerów. W targach biorą udział wszystkie liczące się w kraju firmy softwarowe. Możliwość nabycia oryginalnych programów pozwoli Państwu uatrakcyjnić walory posiadanego sprzętu. Na miejscu można będzie nabyć KATALOG oferowanych programów oraz materiały sympozjalne.

Targi czynne będą w godz. 10.00–17.00.

Przewidywane są następujące formy prezentacji:

wystawa ★ seminaria ★ pokazy promocyjne wystawców.

W ramach Targów prowadzony jest KONKURS na najlepszy produkt programowy.

Miejsce Targów:

OŚRODEK POSTĘPU TECHNICZNEGO
ul. Bytkowska 1b
40-955 KATOWICE

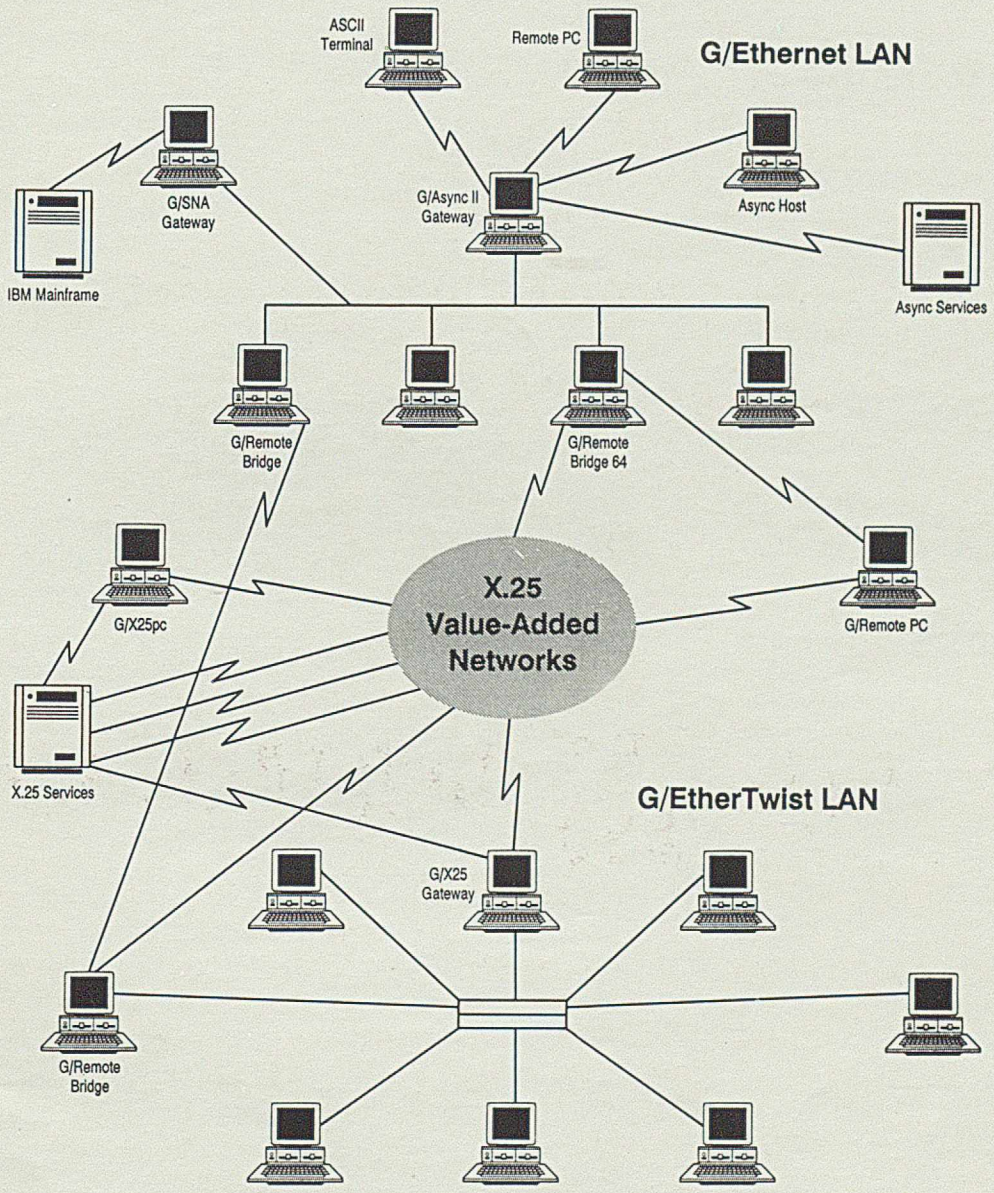
Informacje: **tel. 1541-134, 596-061 (7) wewn. 264, 150**
teleks: 0312458 OPT PL, faks: 588-919

0/3/92

Iszkowski W.: Informatyzacja sukcesem biznesu INFORMATYKA 1992, nr 8, s. 2 Geneza, rozwój oraz przyszła rola technologii informacyjnej z punktu widzenia potrzeb społeczeństwa oraz efektywnej działalności gospodarczej.	**Iszkowski W.: Informatization is a business success** INFORMATYKA 1992, No. 8, p. 2 Genesis, development and future role of information technology from the point of view of society and effective economic activity.	**Iszkowski W.: Informatisation macht Geschäftserfolg** INFORMATYKA 1992, Nr. 8, S. 2 Genesis, Entwicklung und künftige Rolle der Informationstechnologie vom Gesichtspunkt der Gesellschaftsbedürfnisse und der effektiven ekonomischen Tätigkeit.
Piórkowska T., Puternicki D., Staniszkis W.: Inżynieria systemów informatycznych – od strategii do realizacji INFORMATYKA 1992, nr 8, s. 9 Charakterystyka metodyki projektowania i realizacji systemów informatycznych, stosowanej i rozpowszechnianej przez firmę ZETO-RODAN z Warszawy.	**Piórkowska T., Puternicki D., Staniszkis W.: Data processing systems engineering – from strategy to realization** INFOMATYKA 1992, No. 8, p. 9 Characteristics of data processing systems designing and realization methodology, which is applied and distributed by ZETO-RODAN company from Warsaw.	**Piórkowska T., Puternicki D., Staniszkis W.: Ingenieurwesen der EDV-Systeme – von Strategie bis Realisierung** INFORMATYKA 1992, Nr. 8, S. 9 Eine Charakteristik von Methodik der EDV-Systemprojektierung und -realisierung, die von der Firma ZETO-RODAN aus Warschau angewendet und verbreitet wird.
Madrjas K.: AD/Cycle. IBM-owska filozofia inżynierii projektowania oprogramowania INFORMATYKA 1992, nr 8, s. 18 Charakterystyka założeń filozofii firmy IBM w zakresie utrzymania spójności oprogramowania użytkowego dla głównych platform sprzętowych tej firmy.	**Madrjas K.: AD/Cycle. IBM's philosophy of software design engineering** INFORMATYKA 1992, No. 8, p. 18 Characteristics of the IBM philosophy principles to achieve the consistency of application software for main hardware platforms of the company.	**Madrjas K.: AD/Cycle. Eine IBM-Philosophie von Ingenieurwesen der Softwareprojektierung** INFORMATYKA 1992, Nr. 8, S. 18 Eine Charakteristik von Voraussetzugen der IBM-Philosophie im Bereich der Einigkeitssicherung von Anwendungssoftware für leitende Hardwareplattformen dieser Firma.

Produkty sieciowe LAN i WAN

Gateway communications, inc.



BEZKONKURENCYJNE PRODUKTY
SIECI KOMPUTEROWYCH NAGRADZANE PRZEZ:

PC MAGAZINE, LAN MAGAZINE, INFO WORLD

OFERUJE AUTORYZOWANY DYSTRYBUTOR:

MIKROB

PC MAGAZINE EDITORS' CHOICE January 31, 1989 G/Ethernet AT

PC MAGAZINE EDITORS' CHOICE October 16, 1990 G/EtherTwist

P.W.P.T. MIKROB SP. Z O.O.

20-346 Lublin, ul. Długa 5

Tel. (0-81) 420-61, faks 415-43, teleks 643776 MIKRO PL

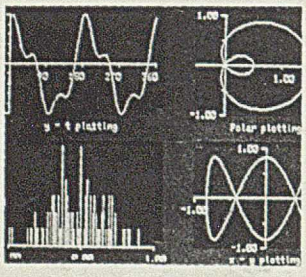
0/15/91

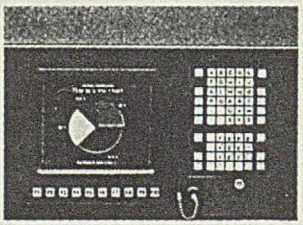
Na życzenie wysyłamy katalog produktów. Atrakcyjny program współpracy dla dealer'ów.

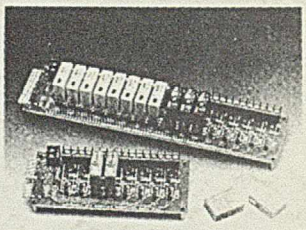
Nowa technologia jest tania ...

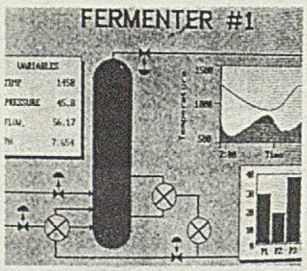
Nie opłaca się robić sterowania czy pomiarów starymi metodami.
Teraz należy korzystać z kart przetworników analogowo cyfrowych,
komputerów przemysłowych, modułów pomiarowych i sterujących.
Zebrane dane możesz przesyłać na dowolne odległości.
Możesz je poddać wyrafinowanej obróbce wykorzystując procesory sygnałowe.
A stosując nowoczesne narzędzia programowe do tworzenia konkretnych
aplikacji otrzymujesz gotowy produkt w kilka dni, jeśli nie godzin !

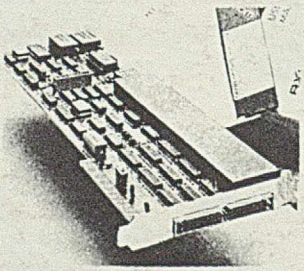
... wystarczy mieć gotowe klocki

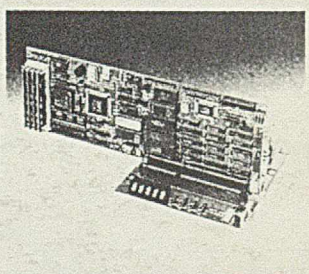


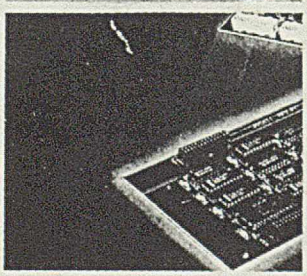


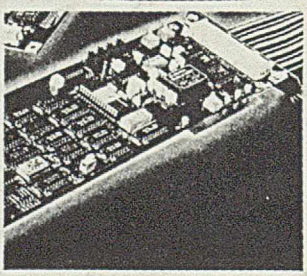


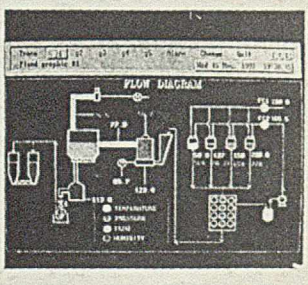












CONVERT®
LABORATORIES

PUIiK "CONVERT"
50-412 WROCŁAW, ul. Mazowiecka 17
tel: (071) 44-85-66, fax: (071) 334-16

Zapraszamy na SOFTARG'92, Katowice, 15-18 września, paw.8

500/92